

Bedienungsanleitung A10-RTK

Erstellungsdatum: 24. Mai 2023
Revision: 1.1



Alberding GmbH

Ludwig-Witthöft-Str. 14
info@alberding.eu



D-15745 Wildau
www.alberding.eu



Bearbeitungshistorie

Version	Datum	Beschreibung der Änderungen
0.9	04.12.2018	Vorabversion
1.0	13.03.2023	Kapitel 4 und 9 hinzugefügt, Sicherheitshinweise aktualisiert
1.0	27.03.2023	Hinweis auf GNSS-Anschlüsse aktualisiert
1.1	24.05.2023	Kapitel 7.2.1 und 7.2.2 hinzugefügt, Kapitel 6.1 - Zugriff auf Webinterface ergänzt, Kapitel 8 - Bearbeitung von Dateien ergänzt



Inhaltsverzeichnis

1	Sicherheitshinweise	4
2	Systemspezifikationen	5
2.1	Systemansatz A10-RTK	5
2.2	A10-RTK Sensor mit GNSS-Board und Embedded-PC	7
2.3	LED-Anzeigen	9
2.4	Pinbelegung der Schnittstelle Sub-D HD26	10
2.5	Technische Spezifikationen A10-RTK	11
3	Inbetriebnahme	13
4	Septentrio Mosaic-X5 GNSS-Board	14
4.1	Kurzbeschreibung	14
4.2	Technische Spezifikationen	14
5	Embedded-PC im A10-RTK Sensor	16
6	Verbindungen	19
6.1	Kabelverbindungen	19
6.2	Bluetooth-Verbindung	26
7	Konfiguration	27
7.1	Konfiguration ohne Linux-Board	27
7.2	Konfiguration mit Linux-Board	27
7.2.1	Einstellung einer statischen IP-Adresse	27
7.2.2	Ports	31
8	Datenverwaltung	31
9	SSRZ mit dem Alberding A10-RTK	33
10	Haftungsausschluss	37



1 Sicherheitshinweise

Warnung!

Bei der Verwendung von elektronischen Geräten müssen stets grundlegende Sicherheitsmaßnahmen befolgt werden, um das Risiko von Bränden, Stromschlägen und Personenschäden zu verringern.

1. Lesen Sie vor der Anwendung des Geräts die vorliegende Bedienungsanleitung aufmerksam durch.
2. Reinigen Sie nur mit einem trockenen Tuch.
3. Installieren Sie gemäß den Anweisungen des Herstellers.
4. Installieren Sie das Gerät nicht in der Nähe von Wärmequellen wie Radiatoren, Heizregister, Öfen oder anderen Apparaturen (einschließlich Verstärker) die Wärme erzeugen.
5. Das Gerät nicht erhitzen oder in ein Feuer werfen.
6. Verwenden Sie ausschließlich vom Hersteller empfohlene Kabel, Befestigungen und Zubehörteile.
7. Das nicht autorisierte Öffnen des Gehäuses des *A10-RTK* Sensors führt zum Verlust des Garantieanspruchs.
8. Achten Sie darauf, dass keine Flüssigkeiten auf oder in das Gerät gelangen.
9. Power over Ethernet (PoE) wird nicht unterstützt. Die Stromversorgung hat ausschließlich über die Power-Schnittstelle zu erfolgen.



2 Systemspezifikationen

2.1 Systemansatz A10-RTK

Die Alberding Telemetrie- und Positionierungssysteme *A07-RTK* und *A10-RTK* wurden für Aufgaben der präzisen Positionierung sowie der Erfassung und Übertragung von Daten konzipiert. Beide Sensoren vereinen zentimetergenaue GNSS RTK-Boards, Mobilfunkmodule und programmierbare Rechner in einem Gehäuse. Die Vorkonfiguration der Funktionsweise und des Datenflusses über die Konfigurator-APP ermöglicht einen autarken Betrieb der „intelligenten Sensoren“ im Feld. In Verbindung mit den Cloud-Lösungen der Alberding GmbH können die mobilen Systeme für unterschiedliche Server-/Client-Anwendungen der Digitalisierung und Automatisierung angepasst werden.



Der Alberding *A10-RTK* Sensor ist eine Weiterentwicklung des Einstiegsmodells *A07-RTK*, der erfolgreich in unterschiedlichen Marktsegmenten (Agrar, Forst, Fahrzeugtracking, GIS, Vermessung, etc.) eingesetzt wird. Das Herzstück des Systems ist ein integrierter Prozessor mit der Alberding Datenmanagementsoftware, welche die Sensorkomponenten und den internen Datenfluss steuert. Die gewünschten Einstellungen werden an der Konfigurator-APP vorgenommen und an die Sensoren übertragen, so dass diese beim Hochfahren (Knopfdruck oder Stromzufuhr) automatisch mit der eingestellten Konfiguration arbeiten.



Im Vergleich zum *A07-RTK* wurde im *A10-RTK* Sensor ein LTE-Modem und ein skalierbares GNSS-Board integriert. Das LTE-Modem bietet gegenüber dem GPRS-Modem im *A07-RTK* eine höhere Datenrate und eine zukunftssichere Perspektive für die mobile Kommunikation mit dem Server.

Zusätzliche Flexibilität und Funktionalität erhält der *A10-RTK* Sensor durch die Integration eines Embedded-PC's mit Linux-Betriebssystem. Während der integrierte Cortex-Prozessor das Datenmanagement übernimmt, kann auf dem Linux-PC im Sensor eine Applikationssoftware laufen, so dass die Anbindung eines externen Rechners entfallen kann.

A10-RTK Systemmerkmale:

- Verwendung skalierbarer GNSS-Hardware (von L1-RTK bis L1/L2-RTK + Heading)
- Integrierte Speicherung von RTK-Positionen und Rohdaten
- Server-/Client-Kommunikation (LTE-Modul)
- Integriertes Bluetooth-Modul
- Integrierter Cortex-Prozessor für das Datenmanagement
- Embedded-PC mit Linux-Betriebssystem (optional)
- Integration von Applikationssoftware
- Flexible Systemanpassung
- Automatisierter Datenfluss
- Autonomer Betrieb durch Vorkonfiguration



2.2 A10-RTK Sensor mit GNSS-Board und Embedded-PC

In der aktuellen Version wird der Alberding *A10-RTK* Sensor mit einem Septentrio Mosaic-X5 GNSS-Board und einem Embedded-PC mit Linux-Betriebssystem geliefert. Die Performance der Positionierung hängt unter anderem von der Konfiguration des Septentrio Boards (vgl. Kap. 4), der verwendeten GNSS-Antenne und den Messbedingungen ab. Die Funktionsweise des Systems wird durch die Vorkonfiguration des Mikrocontrollers (Konfigurator-APP) und die Software auf dem Embedded-PC definiert.

Im Blockdiagramm (vgl. Abb. 1) des *A10-RTK* Sensors sind die wesentlichen Systemkomponenten und Verbindungen dargestellt. Die GNSS-Antenne wird über ein Antennenkabel mit TNC- oder SMA-Stecker und die GSM-Antenne über einen SMA-Anschluss mit dem *A10-RTK* Sensor verbunden. Um den Einbau des Systems zu erleichtern, befinden sich die Verbindungen auf einer Seite des *A10-RTK* Sensors. Im Blockdiagramm wurden die Verbindungen zu den Antennen aus Darstellungsgründen auf die andere Seite gezeichnet.

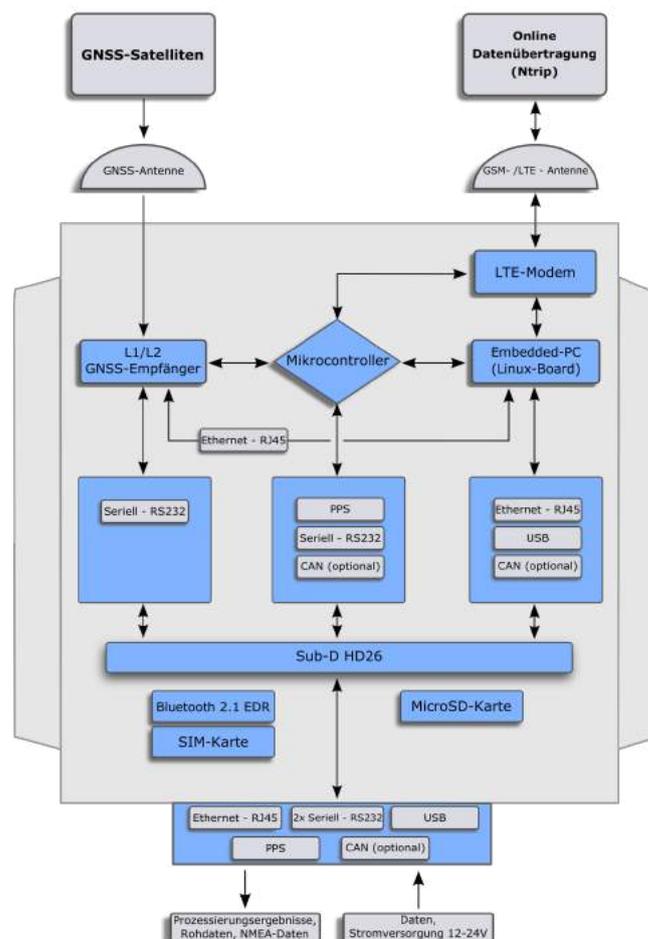


Abbildung 1: Blockdiagramm A10-RTK mit Linux-Board



Da bei vielen Applikationen (z. B. Maschinenanwendung) eine externe Stromversorgung zur Verfügung steht, wurde beim *A10-RTK* auf die Integration eines Lithium-Polymer-Akkus verzichtet. Die Stromversorgung des Sensors (12-24V DC) erfolgt gemeinsam mit der kabelgebundenen Kommunikation über den 26-poligen Stecker des *A10-RTK*. An diesem Port stehen 1 Ethernet-Anschluss, 2 serielle RS232- und 1 USB-Schnittstellen sowie optional ein CAN-Port zur Verfügung.

Auf der Oberseite des *A10-RTK* Sensors befinden sich der An- und Ausschalter, die *Log*-Taste und die fünf Status-LEDs. Die *Log*-Taste ist momentan mit keiner Funktion belegt. An der Frontseite stehen Ihnen folgende Anschlüsse zur Verfügung: GNSS- und GSM-Antenne, sowie eine serielle Schnittstelle (Sub-D HD26) (vgl. Abb. 2). Die serielle Schnittstelle wird werkseitig mit 115200 Baud 8-N-1 konfiguriert.

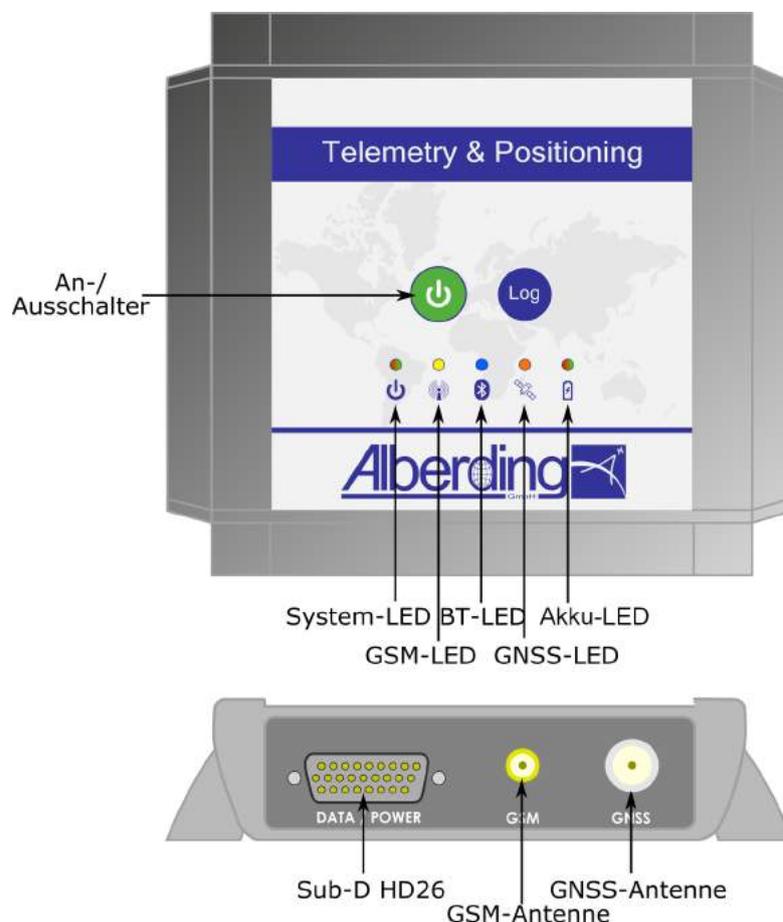


Abbildung 2: Schematischer Aufbau des A10-RTK



2.3 LED-Anzeigen

Folgend werden die LED-Anzeigen vom *A10-RTK* erläutert (vgl. Abb. 3).

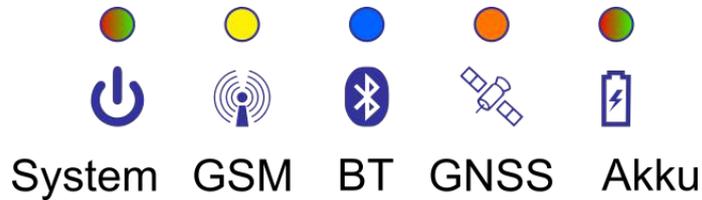


Abbildung 3: LED-Anzeige

Nach Erreichen der Betriebsbereitschaft:

LED	Farbe und Eigenschaft	Erklärung
LED 1 System	Grün/Rot blinkend	System läuft
LED 2 GSM	Aus Gelb blinkend alle 1s blinkend alle 3s blinkend alle 0,5s	Funkmodem deaktiviert Initialisierung Einwahlvorgang Datenübertragung aktiv SIM Error
LED 3 Bluetooth	Aus Blau blinkend alle 1s blinkend alle 3s	Bluetooth deaktiviert Initialisierung Bereit zum Koppeln Datenübertragung aktiv
LED 4 GNSS	Aus Orange blinkend alle 1s blinkend alle 3s doppel blinken alle 3s	GNSS deaktiviert Initialisierung Standalone RTK-Float RTK-Fix
LED 5 Akku	/	Momentan nicht belegt

2.4 Pinbelegung der Schnittstelle Sub-D HD26

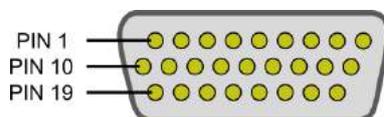


Abbildung 4: Pinbelegung der Schnittstelle Sub-D HD26

PIN	Beschreibung	Kommentar
1	reserviert	/
2	CAN L	wahlweise Linux-Board oder Mikrocontroller (optional)
3	reserviert	/
4	reserviert	/
5	reserviert	/
6	GND	/
7	TX3 (RS232) zum GNSS-Board	Direktverbindung zum GNSS-Board
8	RX3 (RS232) zum GNSS-Board	Direktverbindung zum GNSS-Board
9	USB D+	Standardmäßig Linux-Board; Optional GNSS-Board
10	reserviert	/
11	CAN H	wahlweise Linux-Board oder Mikrocontroller (optional)
12	TX2 (RS232)	Mikrocontroller (Datenverteilung konfigurierbar)
13	reserviert	/
14	reserviert	/
15	reserviert	/
16	Netzwerk (RJ45) RX-	Linux-Board / GNSS-Board
17	Netzwerk (RJ45) TX-	Linux-Board / GNSS-Board
18	USB D-	Standardmäßig Linux-Board; Optional GNSS-Board
19	reserviert	/
20	GPIO	Mikrocontroller (optional)
21	RX2 (RS232)	Mikrocontroller (Datenverteilung konfigurierbar)
22	reserviert	/
23	Power GND	/
24	Power IN DC	DC 12-24V
25	Netzwerk (RJ45) RX+	Linux-Board / GNSS-Board
26	Netzwerk (RJ45) TX+	Linux-Board / GNSS-Board



2.5 Technische Spezifikationen A10-RTK

Datenfluss und Speicherung

Datenspeicherung:	Integrierte Speicherkarte
Datenausgang kabellos:	Bluetooth, mobiles Internet
Dateneingang kabellos:	Bluetooth, mobiles Internet
Datenausgang kabelgebunden:	LAN, Seriell RS232, Seriell USB, PPS out
Dateneingang kabelgebunden:	LAN, Seriell RS232, Seriell USB, Event In

Kommunikation

Bluetooth 2.1 + EDR:	Klasse 2, Reichweite: ~5 m, SPP-Protokoll
Funk:	LTE (800/850/900/1800/2100/2600 MHz)
Data/Power:	Sub-D HD26

Physikalische Eigenschaften

Abmessung (LxBxH):	13.0 cm x 15.3 cm x 3.3 cm (5.12" x 6.02" x 1.30")
Gewicht:	656g (1,45lb)
Statusanzeigen (LEDs):	System, GSM, Bluetooth, GNSS-Status, Ladestatus
Bedienelemente:	Power, Log
Antennenanschluss:	GNSS: TNC- oder SMA-Buchse GSM: SMA-Buchse

Umweltverträglichkeit

Betriebstemperatur:	-20°C to +55°C (-4°F to +131°F)
Rel. Luftfeuchtigkeit:	bis 80%
Schutzklasse:	IP67
Gehäusematerial:	Metall (AlMgSi 0.5) pulverbeschichtet
Konformität:	CE, RoHS und bleifrei



Elektrische Eigenschaften

Externe Stromversorgung:	12 - 24 V DC
Stromverbrauch:	Konfigurationsabhängig, bis zu 5 W mit Linux-Board und mobilen Internet (Datenstreaming)
Sicherheitsschutz:	Kurzschluss Strombegrenzung Überspannung Temperaturüberwachung

Optionales Zubehör

Multiport-Adapter (LAN, RS232, Power)

Netzteil mit Netzkabel

Octopus Kabel (LAN, RS232-Buchse, RS232-Stecker, PSS out, Event In, USB, Power)

Externe GNSS-Antenne

Externe GSM-Antenne



3 Inbetriebnahme

Für die Inbetriebnahme wird der *A10-RTK* Sensor extern mit Strom versorgt. Im Standardlieferungsumfang ist dazu ein Netzteil mit Netzkabel und der Multiportadapter enthalten (vgl. Abb. 5).

Verbinden Sie das Netzteil mit dem Multiportadapter und stecken Sie diesen an die 26-polige Schnittstelle (Sub-D HD26) des *A10-RTK* (vgl. Abb. 2). Beim Anliegen der Stromversorgung blinken alle LED's auf. Der *A10-RTK* Sensor wird entweder durch das Drücken des An-/Ausschalters oder die Aktivierung der Auto-Power-On Funktion gestartet. Die blinkende System-LED signalisiert den Startvorgang, der nach ca. 30 Sekunden abgeschlossen ist.

Schließen Sie die GNSS-Antenne (TNC- oder SMA-Stecker) und die GSM-Antenne (SMA-Stecker) an den Sensor an (vgl. Abb. 2).

Achtung: Der *A10-RTK* führt während des Starts automatisch eine Pin-Abfrage der SIM-Karte durch. Vor der ersten Inbetriebnahme sind daher die verschiedenen Mobilfunkeinstellungen (SIM-Pin, Server, etc.) mit der Konfigurator-APP vorzunehmen.

Die eingestellte Konfiguration (Mobilfunkeinstellungen, Datenfluss) wird im *A10-RTK* gespeichert, so dass der Sensor bei der erneuten Inbetriebnahme automatisch mit diesen Einstellungen arbeitet. Durch das Drücken des An-/Ausschalters wird der *A10-RTK* ausgeschaltet.



Abbildung 5: A10-RTK mit Multiport-Adapter, Netzteil, GSM- und GNSS-Antenne (Standardlieferungsumfang)



4 Septentrio Mosaic-X5 GNSS-Board

4.1 Kurzbeschreibung

Das Septentrio Mosaic-X5 Board verfügt über 448 Empfangskanäle für die Verfolgung von bis zu 6 GNSS-Systemen (GPS, GLONASS, Galileo, BeiDou, QZSS, Navic) auf drei Frequenzen (L1, L2, L5). Der Ethernet-Port des Mosaic-X5 ist über die Ethernet-Schnittstelle des Linux-PC's im *A10-RTK* erreichbar. Die Inbetriebnahme und Konfiguration des GNSS-Boards erfolgt über das integrierte Web-Interface. In verschiedenen Videos beschreibt die Firma Septentrio auf ihrer Homepage die Nutzung des Web-Interfaces (Septentrio Mosaic-X5 Dokumentationen).

Die RTK-Korrekturdaten können dem Mosaic-X5 Board im *A10-RTK* kabelgebunden über die serielle Schnittstelle oder über Ethernet-Port bzw. kabellos über die mobile Internetverbindung (Ntrip) oder Bluetooth zugeführt werden. Diese Verbindungen werden auch für die Bereitstellung der präzisen Positionsdaten (NMEA-Format) genutzt.

4.2 Technische Spezifikationen

Tracking skalierbar

GNSS Signale:

GPS	L1C/A, L1PY, L2C, L2P, L5
GLONASS	L1CA, L2CA, L2P, L3 CDMA
BeiDou	B1I, B1C, B2a, B2I, B3
Galileo	E1, E5a, E5b, E5 AltBoc
QZSS	L1C/A, L2C, L5
Navic	L5
SBAS	Egnos, WAAS, GAGAN, MSAS, SDCM (L1, L5)
Kanäle:	448
Wiederholrate:	RTK: bis zu 100 Hz

Genauigkeit (RMS)¹

RTK horizontale Genauigkeit:	0.060 m + 0.5 ppm
RTK vertikale Genauigkeit:	0.010 m + 1 ppm
RTK Initialisierungszeit:	7 s

¹Abhängig von Basislinienlänge (< 40 km), Anzahl an sichtbaren Satelliten, Satellitengeometrie, GNSS Antenne, Mehrwegeeffekten und atmosphärischen Störungen



Time to First Fix

Kaltstart:	< 45 s
Warmstart:	< 20 s
Zeit bis zum Wiedererlangen:	1 s

E/A Schnittstelle

USB:	USB device (2.0, HS)
E/A:	1 PPS out / Event In
LAN:	HTTP (web GUI), NTP Server, NtripCaster, Dynamisches DNS
Serial:	2 UART

Datenfluss

Datenausgang in Echtzeit:	NMEA-0183, RTCM-Version 3.x, RINEX-Version 3.x, SBF ² , CMR v2.0
Dateneingang in Echtzeit:	RTCM-Version 3.x, CMR v2.0, CMR+, SBF

²SBF: Septentrio Binary Format



5 Embedded-PC im A10-RTK Sensor

Die Integration eines Embedded-PC's im *A10-RTK* Sensor ist die Voraussetzung für eine flexible Systemanpassung auf unterschiedliche Aufgaben der Digitalisierung und Automatisierung. Während der integrierte Mikroprozessor den Datenfluss im *A10-RTK* Sensor regelt, kann die Software auf dem Embedded-PC umfangreichere Berechnungen durchführen. Aktuell werden sowohl Softwarelösungen der Alberding GmbH (z. B. EuroRef, EuroNet) als auch Fremdsoftware (z. B. BNC vom Bundesamt für Kartographie und Geodäsie, SSR2OSR-Konverter der Firma Geo++ GmbH) unterstützt.

Beispiele für Softwareapplikationen auf dem integrierten Embedded-PC:

- Autark operierende GNSS-Referenz- und Monitoringstation
 - Bereitstellung von RTK-Korrekturdaten und GNSS-Rohdaten
 - Überwachung von RTK-Diensten im Feld

- Erweiterte GNSS-Roverfunktionalität
 - Unterstützung von 2 unterschiedlichen IP-Adressbereichen für die RTK-Korrekturdaten und die GNSS-Positionsdaten (vgl. Abb. 6)
 - Umschalten zwischen 2 Korrekturdatenquellen (z. B. mobiles Internet und Datenfunk im 70cm Band)

- Integration von GNSS-Algorithmen
 - GNSS-Datenkonvertierung (SSR->OSR, DataConv)
 - GNSS-Positionierungsalgorithmen (BNC, RTKLib)

- Überwachung von Objekten, Bauwerken und Böschungen
 - Ingenieurtechnisches Geomonitoring
 - Geofencing, Überwachung von Maschinen

- Sensorfusion
 - Kombinierte Erfassung von GNSS- und externen Sensoren (EuroNet)
 - Kombinierte Auswertung von GNSS- und externen Sensordaten



- Projektdatenübertragung vom und zum Server
 - Übertragung von ingenieurtechnischen Projektdaten (digitales Geländemodell, Stationierungskoordinaten)
 - Übertragung von Bewirtschaftungsdaten (teilflächenspezifische Düngung)

- Telemetrie-Modul mit RTK
 - Auslesen und Übertragung der Daten von Land- und Baumaschinen
 - Kombination von Telemetrie und RTK

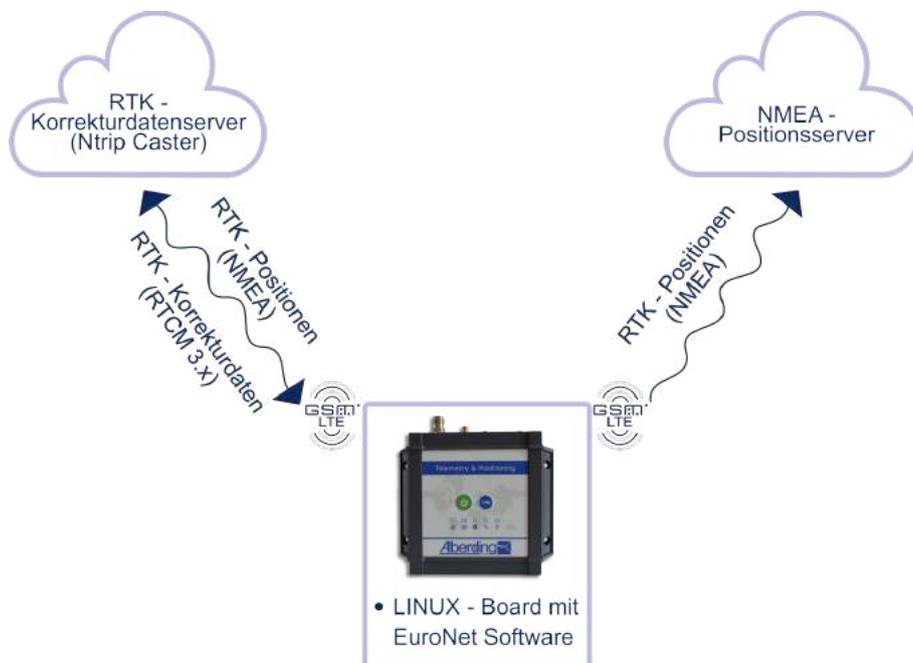


Abbildung 6: Unterstützung von 2 unterschiedlichen IP-Adressbereichen



Abbildung 7: Beispiel für eine Maschineninstallation



6 Verbindungen

6.1 Kabelverbindungen

Seriell:

Wenn in Ihrem PC keine serielle Schnittstelle verbaut ist, müssen Sie einen USB-RS232 Adapter verwenden. Achten Sie beim Kauf darauf, dass der USB-RS232 Adapter mit ihrem Betriebssystem (z. B. Windows 10) kompatibel ist und, dass der Hersteller passende Treiber dafür anbietet.

Damit der USB-RS232 Adapter (PC) mit dem Multiport-Adapter (A10-RTK) verbunden werden kann, müssen ein Gender-Changer und ein Null-Modem Adapter verwendet werden.

Verbinden Sie alle Kabel und schließen Sie diese an den PC an. Den passenden COM-Anschluss finden Sie bei Windows-Betriebssystemen wie folgt:

Windows 7:

Systemsteuerung > Geräte-Manager > Anschlüsse (COM & LPT)

Windows 10:

Einstellungen > Geräte > Geräte-Manager > Anschlüsse (COM & LPT)

Ethernet:

Es wird ein Ethernet-Kabel benötigt, das direkt mit dem Computer verbunden ist.

Die IP-Adresse des A10-RTK Sensors wird von der Alberding GmbH festgelegt und setzt sich aus der Seriennummer des A10-RTK zusammen:

Beispiele:

S/N A10-RTK	IP-Adresse (Alberding GmbH)
A10-50010	10.50.0.10
A10-50110	10.50.1.10
A10-50223	10.50.2.23

Tabelle 3: IP-Adressenvergabe (Alberding GmbH)

Unabhängig davon müssen Sie eine zweite IP-Adresse im gleichen Subnetz vergeben. Dazu müssen die ersten drei Bereiche der IP-Adresse identisch sein.

**Beispiele:**

S/N A10-RTK	IP-Adresse (Alberding GmbH)	IP-Adresse (Kunde)
A10-50010	10.50.0.10	10.50.0.101
A10-50110	10.50.1.10	10.50.1.101
A10-50223	10.50.2.23	10.50.2.101

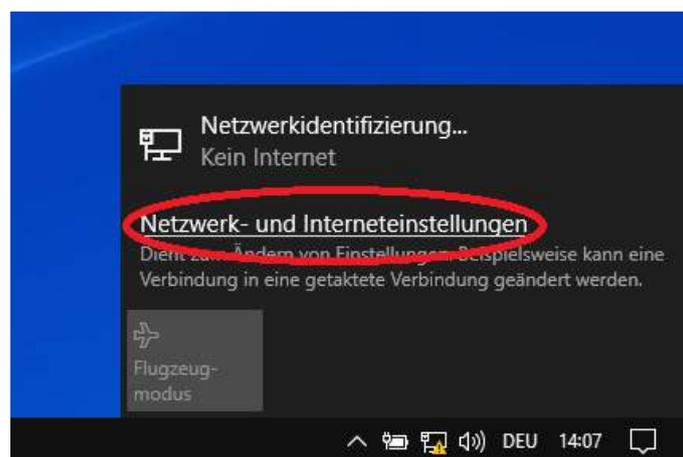
Tabelle 4: IP-Adressenvergabe (Alberding GmbH - Kunde)

Diese Vergabe kann der Netzwerkadministrator vornehmen oder Sie befolgen folgende Schritte unter **Linux**:

1. Führen Sie das Programm *Terminal* oder *Konsole* aus.
2. `sudo ip addr add <IP-Adresse (Kunde)>/24 dev eth0`
3. Hinweis: Nach dem Ausführen des nächsten Befehls wird ein Passwort abgefragt. Dieses wird Ihnen von der Alberding GmbH zur Verfügung gestellt.
4. `ssh euronav@10.50.X.XX` (IP-Adresse (Alberding GmbH))

Diese Verbindung kann mit den folgenden Schritten unter **Windows 7** oder **Windows 10** hergestellt werden:

1. Klicken Sie mit der linken Maustaste auf das Netzwerksymbol und drücken Sie auf „Netzwerk- und Interneteinstellungen“ (Windows 10), oder auf „Netzwerk- und Freigabecenter öffnen“ (Windows 7) (vgl. Abb. 8 und 9).

**Abbildung 8:** Netzwerk- und Interneteinstellungen (Windows 10)

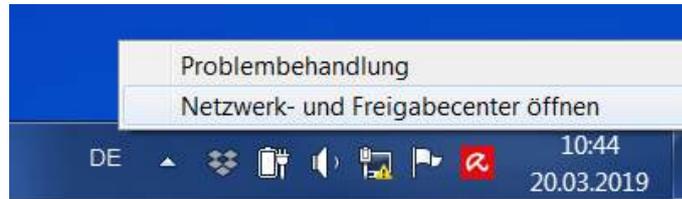


Abbildung 9: Netzwerk- und Freigabecenter öffnen (Windows 7)

2. Klicken Sie dann auf „Adaptoptionen ändern“ (vgl. Abb. 10) in Windows 10 oder auf „Adaptiereinstellungen ändern“ (vgl. Abb. 11) in Windows 7.

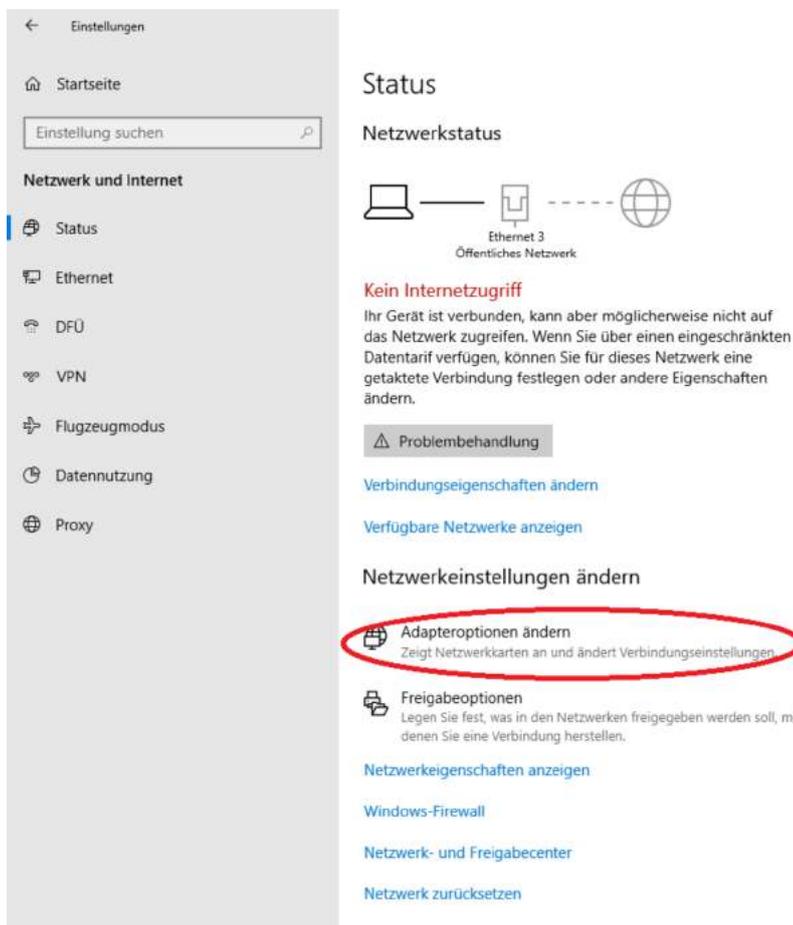


Abbildung 10: Adaptoptionen ändern (Windows 10)

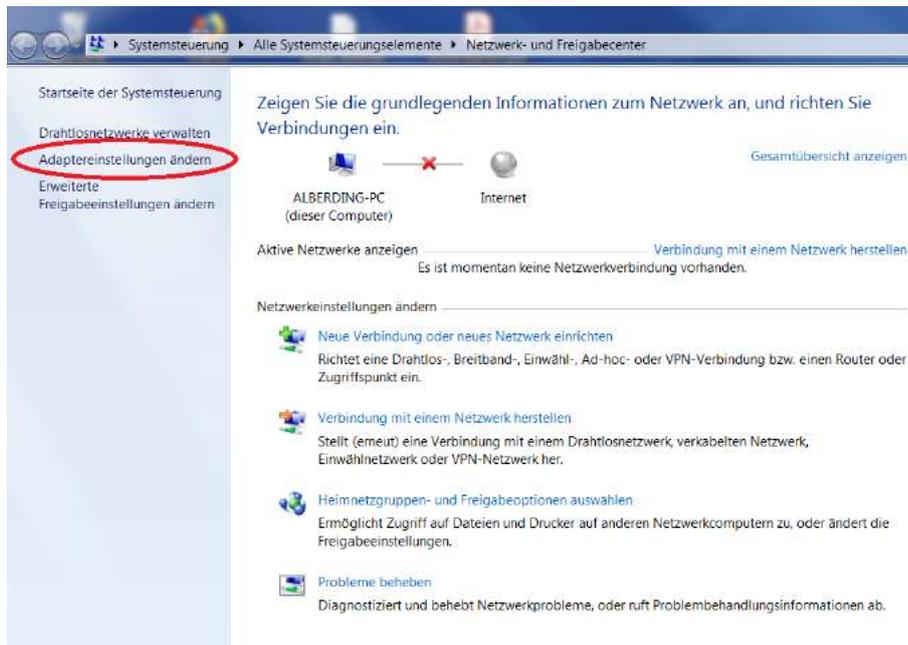


Abbildung 11: Adaptereinstellungen ändern (Windows 7)

3. Es öffnet sich folgendes Fenster (vgl. Abb. 12). Wählen Sie mit der rechten Maustaste die Ethernet-Verbindung aus und klicken Sie auf „Eigenschaften“.

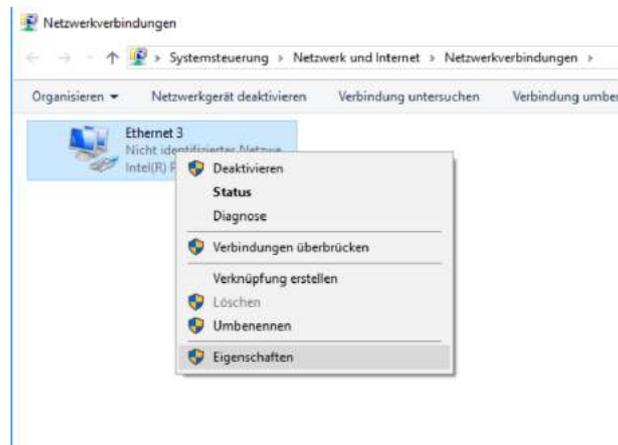


Abbildung 12: Ethernet-Verbindung



4. Wählen Sie „Internetprotokoll, Version 4 (TCP/IPv4)“ aus und drücken Sie auf „Eigenschaften“ (vgl. Abb. 13).

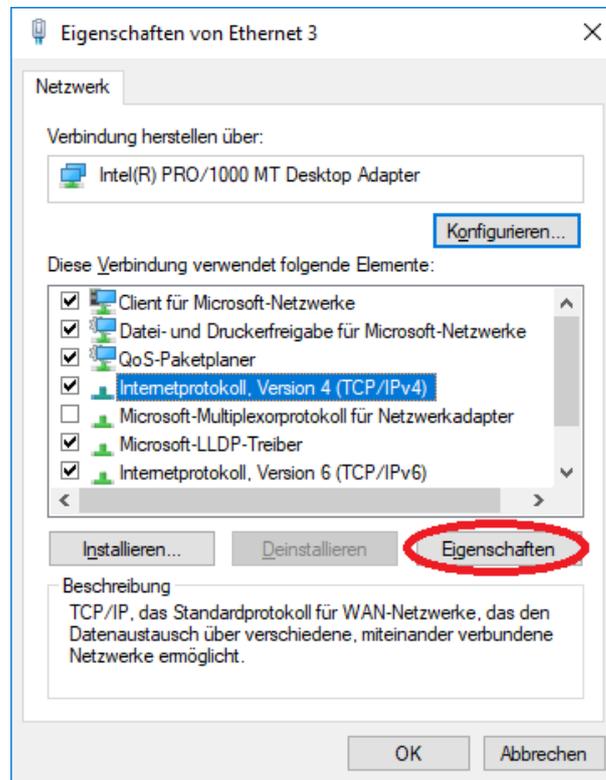


Abbildung 13: Internetprotokoll, Version 4 (TCP/IPv4)



5. Das Fenster „Eigenschaften von Internetprotokoll, Version 4 (TCP/IPv4)“ zeigt die aktuelle Konfiguration an. Bitte machen Sie einen Screenshot oder notieren Sie sich die in diesem Fenster enthaltenen Informationen. Diese benötigen Sie, um die Internetverbindung am Ende der Verbindung mit dem *A10-RTK* Sensor wiederherzustellen (vgl. Abb. 14).

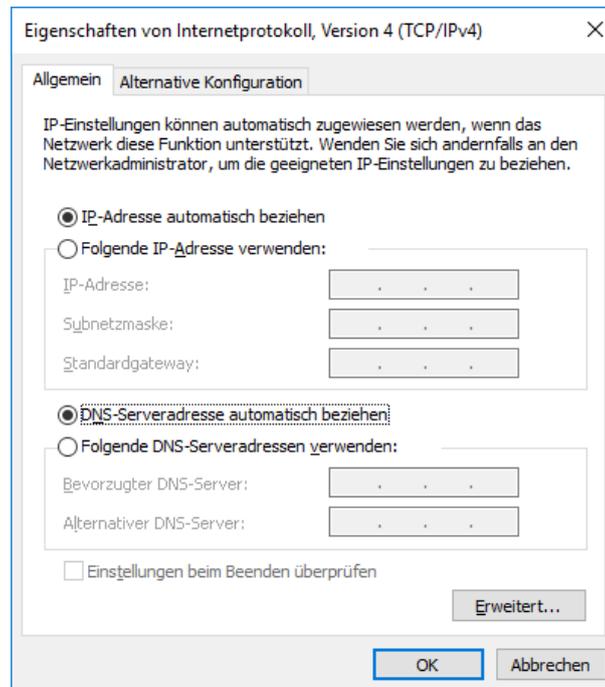


Abbildung 14: Eigenschaften von Internetprotokoll, Version 4 (TCP/IPv4)



6. Geben Sie zum Angeben der IP-Adresse (Kunde) in die Felder IP-Adresse und Subnetzpräfixlänge (255.255.255.0) die Einstellungen für die IP-Adresse ein (vgl. Abb. 15).

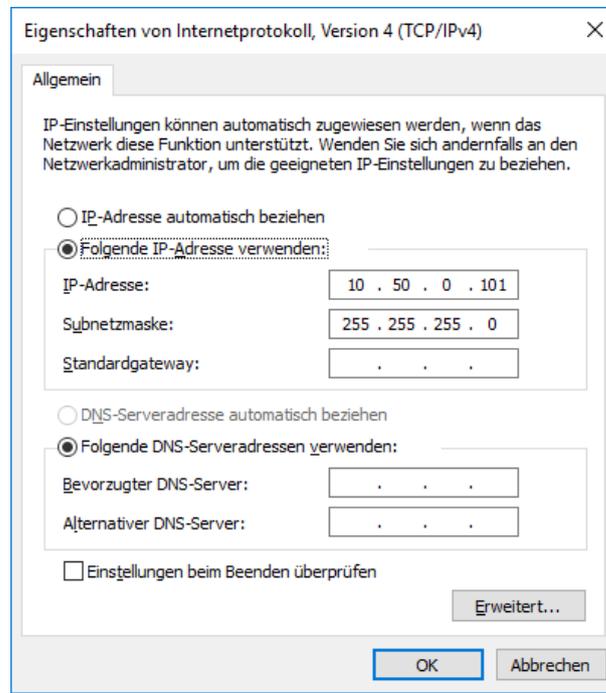


Abbildung 15: IP-Adresse

Durch Betätigen von „OK“ wird die Verbindung zwischen dem *A10-RTK* und dem PC hergestellt.

7. Um die Internetverbindung Ihres Computers wiederherzustellen, wiederholen Sie die Schritte 1 bis 4 und ändern Sie die Informationen im Fenster „Eigenschaften von Internetprotokoll, Version 4 (TCP/IPv4)“ wieder so, wie sie in Schritt 5 waren (vgl. Abb. 14).

Für den Zugriff auf das Webinterface des GNSS-Empfängers muss der Port 8080 verwendet werden. Geben Sie dafür in einen Browser die IP-Adresse und den Port wie folgt ein: 10.50.X.XX:8080.

Achtung: Stellen Sie die interne IP-Adresse des GNSS-Empfängers **nicht** über das Webinterface um. Der GNSS-Empfänger und das Linux-Board können dann nicht mehr miteinander kommunizieren. Führen Sie die IP-Adressen Einstellungen nur mit dem Kommandozeilenprogramm *nmcli* durch (siehe Kapitel 7.2.1).



6.2 Bluetooth-Verbindung

Sobald der *A10-RTK* betriebsbereit ist, wird der *A10-RTK* in der Bluetooth-Umgebung angezeigt. Koppeln Sie den *A10-RTK* mit Ihrem Gerät unter Verwendung der Pin „0000“ (vgl. Bedienungsanleitung Konfigurator.pdf). Der *A10-RTK* installiert sich als serieller Anschluss (COM-Port, 115200, 8-N-1). Es werden keine weiteren Treiber benötigt.

Den passenden COM-Anschluss finden Sie bei Windows-Betriebssystemen wie folgt:

Windows 7:

Systemsteuerung > Geräte und Drucker anzeigen > A10 (Rechtsklick) > Eigenschaften > Dienste

Windows 10:

Start > Einstellungen > Geräte > Weitere Bluetooth-Optionen > COM-Anschlüsse (Reiter) > A10 # 'SPP'

Unterstützte Geräte:

- Geräte ab dem Bluetooth 2.0-Standard
- Apple-Systeme
- Android-Systeme



7 Konfiguration

7.1 Konfiguration ohne Linux-Board

Zum Konfigurieren des *A10-RTK* Sensors mit dem *Alberding Konfigurator* muss der *A10-RTK* über die serielle Schnittstelle oder über Bluetooth (vgl. Kap. 6.2) mit dem PC verbunden werden.

Die Konfigurationssoftware *Alberding Konfigurator* steht unter folgendem Link zum Download bereit und beinhaltet neben dem Konfigurationstool auch den erforderlichen USB-Treiber für Windows-Systeme:

https://www.alberding.eu/programs/A07_Konfigurator.exe

7.2 Konfiguration mit Linux-Board

Um mit dem Linux-Board auf dem *A10-RTK* kommunizieren zu können, muss der *A10-RTK* Sensor via Ethernet mit dem PC verbunden werden. Dann können Programme oder Skripte installiert werden.

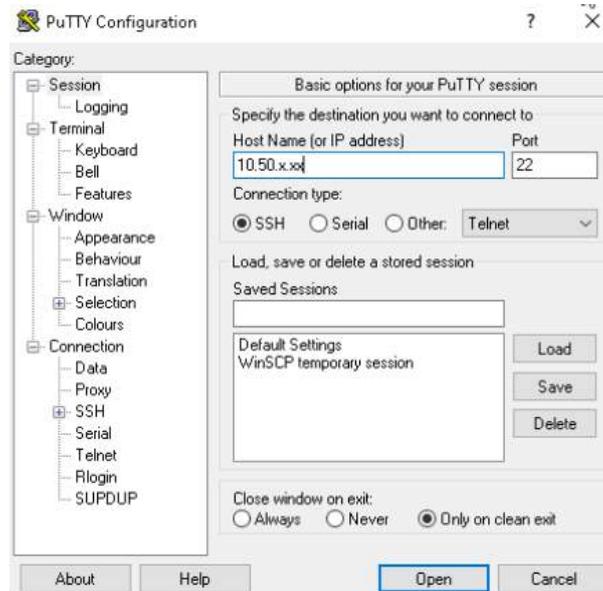
Diese Vergabe kann der Netzwerkadministrator vornehmen oder Sie befolgen folgende Schritte unter **Windows**:

1. Mit der freien Software *PuTTY* kann jetzt eine Verbindung zum *A10-RTK* hergestellt werden: `ssh euronav@10.50.X.XX` (IP-Adresse (Alberding GmbH))
2. Hinweis: Für den Aufbau einer SSH-Verbindung wird ein Passwort angefordert. Dieses wird Ihnen von der Alberding GmbH zur Verfügung gestellt.

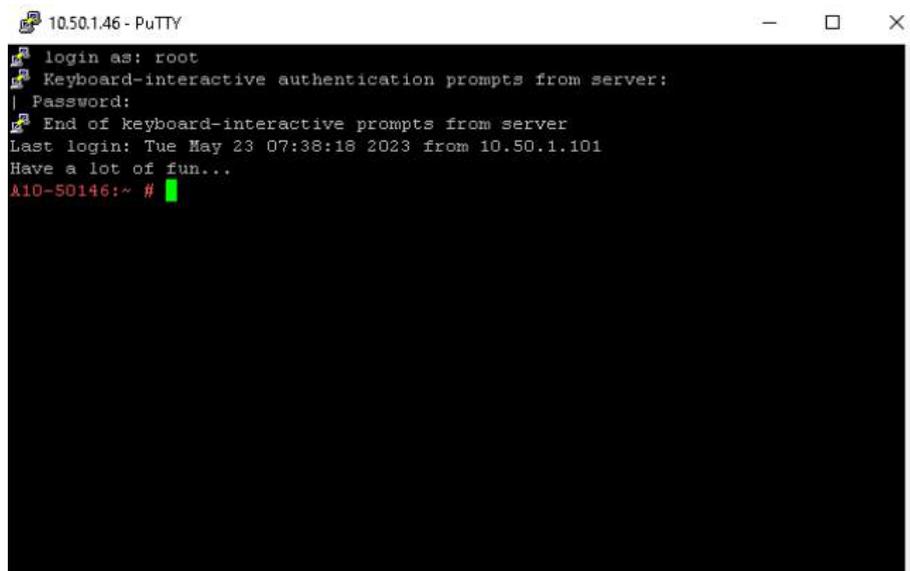
7.2.1 Einstellung einer statischen IP-Adresse

Mit der freien Software *PuTTY* können Sie unter Windows eine statische IP-Adresse vergeben.

1. Öffnen Sie das Programm *PuTTY*. Geben Sie die IP-Adresse (vgl. Tabelle 3) ein und verbinden Sie den A10 via SSH.



2. Nach dem Verbinden öffnet sich das Terminal. Melden Sie sich hier mit dem Benutzer und dem Passwort an, welches Sie von der Alberding GmbH bekommen haben.



3. Mit dem Kommandozeilenprogramm **nmcli** kann eine statische IP-Adresse vergeben werden. Gehen Sie dabei wie folgt vor:
 - a) Anzeigen der Verbindungsprofile mit **nmcli con show**



```

10.50.146 - PuTTY
login as: root
Keyboard-interactive authentication prompts from server:
| Password:
| End of keyboard-interactive prompts from server
Last login: Tue May 23 07:38:18 2023 from 10.50.1.101
Have a lot of fun...
A10-50146:~ # nmcli con show
NAME                UUID                                TYPE      DEVICE
nm_eth0_fb          87890595-0706-49e6-b9aa-f12ead667d9b  ethernet  eth0
nm_eth1             13c52827-b362-4c1f-bf97-ce1c11818f7b  ethernet  eth1
nm_eth0             1a7f20ed-0f84-3edd-b1dc-b5ab14d39de8  ethernet  --
nm_gsm              1efc2732-168c-41ce-9f01-6a76ff52f8d5  gsm       --
A10-50146:~ #

```

- b) Editieren des Profils mit **nmcli con edit nm_eth0** oder **nmcli con edit nm_eth0_fb**
- i. mit **_fb**: feste IP-Adresse ohne DHCP
 - ii. ohne **_fb**: feste IP-Adresse zusätzlich zu der DHCP-Adresse

```

10.50.146 - PuTTY
login as: root
Keyboard-interactive authentication prompts from server:
| Password:
| End of keyboard-interactive prompts from server
Last login: Tue May 23 07:38:18 2023 from 10.50.1.101
Have a lot of fun...
A10-50146:~ # nmcli con show
NAME                UUID                                TYPE      DEVICE
nm_eth0_fb          87890595-0706-49e6-b9aa-f12ead667d9b  ethernet  eth0
nm_eth1             13c52827-b362-4c1f-bf97-ce1c11818f7b  ethernet  eth1
nm_eth0             1a7f20ed-0f84-3edd-b1dc-b5ab14d39de8  ethernet  --
nm_gsm              1efc2732-168c-41ce-9f01-6a76ff52f8d5  gsm       --
A10-50146:~ # nmcli con edit nm_eth0_fb
==== nmcli interactive connection editor |====
Editing existing '802-3-ethernet' connection: 'nm_eth0_fb'

Type 'help' or '?' for available commands.
Type 'describe [<setting>.<prop>]' for detailed property description.

You may edit the following settings: connection, 802-3-ethernet (ethernet), 802-1x, dcb, ipv4, ipv6, tc, proxy
nmcli>

```

- c) Setzen Sie die IP-Adresse mit **set ipv4.addresses xxx.xxx.xxx.xxx/24** und überprüfen Sie die Einstellungen mit **print**.

Informieren Sie sich bei Ihrer IT-Leitung, ob eine Gateway Einstellung notwendig ist. Falls ja, führen Sie folgenden Befehl aus: **set ipv4.gateway xxx.xxx.xxx.xxx**.



```
10.50.1.46 - PuTTY
Type 'help' or '?' for available commands.
Type 'describe [<setting>.<prop>]' for detailed property description.

You may edit the following settings: connection, 802-3-ethernet (ethernet), 802-
ix, dcb, ipv4, ipv6, tc, proxy
nmcli> set ipv4.addresses 111.11.1.111/24
nmcli> print
=====
Connection profile details (nm_eth0_fb)
=====
connection.id:                nm_eth0_fb
connection.uuid:              87890595-0706-49e6-b9aa-f12ead667d9b
connection.stable-id:        --
connection.type:              802-3-ethernet
connection.interface-name:    eth0
connection.autoconnect:       yes
connection.autoconnect-priority: 222
connection.autoconnect-retries: -1 (default)
connection.auth-retries:      -1
connection.timestamp:         1684829555
connection.read-only:         no
connection.permissions:       --
connection.zone:              --
```

- d) Geben Sie anschließend zur Überprüfung **verify** ein und speichern Sie die Einstellungen mit **save**.

```
10.50.1.46 - PuTTY
ipv6.routes:                  --
ipv6.route-metric:           -1
ipv6.route-table:            0 (unspec)
ipv6.ignore-auto-routes:     no
ipv6.ignore-auto-dns:        no
ipv6.never-default:          no
ipv6.may-fail:               yes
ipv6.ip6-privacy:            -1 (unknown)
ipv6.addr-gen-mode:          stable-privacy
ipv6.dhcp-send-hostname:     yes
ipv6.dhcp-hostname:          --
ipv6.token:                  --
-----
proxy.method:                 none
proxy.browser-only:          no
proxy.pac-url:                --
proxy.pac-script:            --
-----
nmcli> verify
Verify connection: OK
nmcli> save
Connection 'nm_eth0_fb' (87890595-0706-49e6-b9aa-f12ead667d9b) successfully updated.
nmcli>
```

- e) Verlassen Sie das Programm nmcli mit **quit**.

Starten Sie anschließend den A10 neu, um die Änderungen zu übernehmen.



7.2.2 Ports

Alle Ports von 2000-10000 sind für das GNSS-Board reserviert. Alle anderen Ports kann das Linux-Board nutzen.

8 Datenverwaltung

Um auf das Dateisystem des Embedded-PC Linux OS zugreifen zu können, muss der *A10-RTK* Sensor über Ethernet mit dem PC verbunden sein. Um Daten vom *A10-RTK* Sensor auf Ihren Computer zu übertragen, verwenden Sie WinSCP und führen Sie diese Schritte aus.

1. Mit der freien Software WinSCP kann jetzt eine Verbindung zum *A10-RTK* hergestellt werden. Stellen Sie bei Übertragungsprotokoll *SFTP* ein. Bei dem Rechnernamen nutzen Sie die IP-Adresse 10.50.X.XX (IP-Adresse - Alberding GmbH) aus der Tabelle (siehe oben). Benutzername und Kennwort werden Ihnen von der Alberding GmbH zur Verfügung gestellt (vgl. Abb. 16).

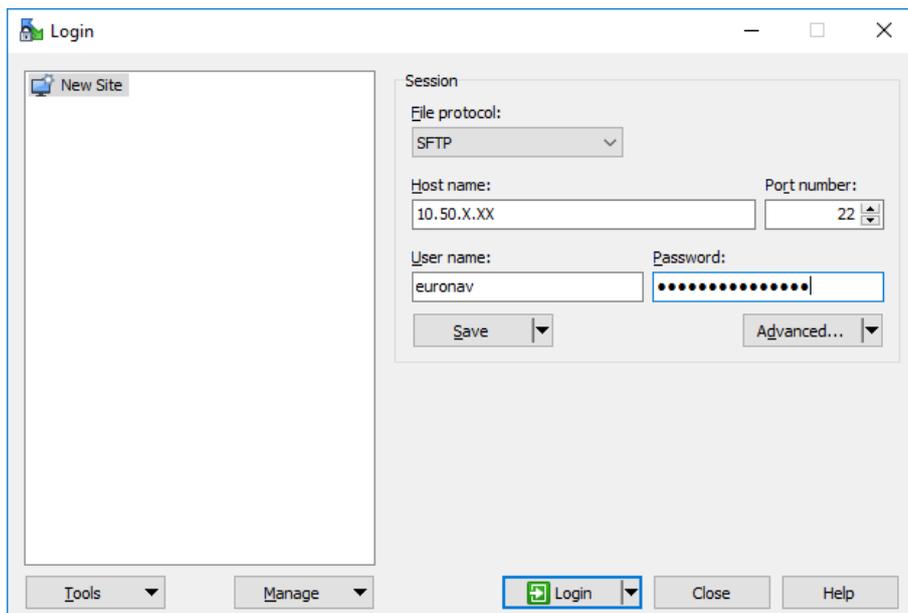


Abbildung 16: WinSCP Konfiguration



- Standardmäßig wird die Verbindung im Ordner „/home/euronav/“ hergestellt. Um auf die Daten des Sensors zuzugreifen, gehen Sie zwei Ebenen nach oben, indem Sie auf „..“ klicken (vgl. Abb. 17).

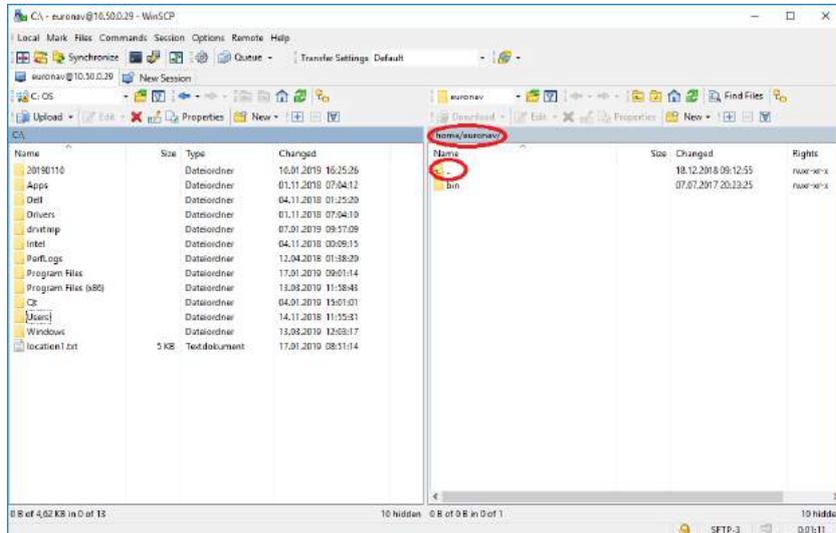


Abbildung 17: WinSCP Verbindung

- Navigieren Sie in WinSCP zum „/var/euronav/data“-Verzeichnis des A10-RTK (vgl. Abb. 18).

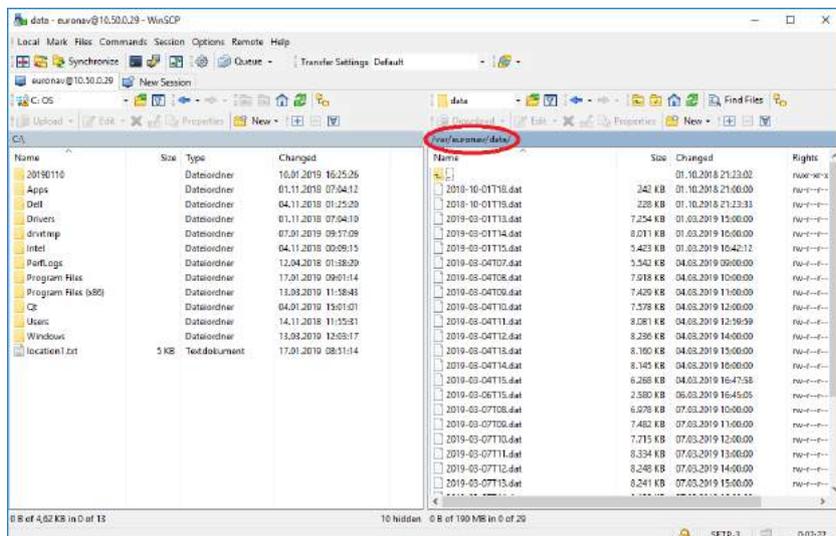


Abbildung 18: WinSCP Daten-Verzeichnis

Jetzt können Sie die Daten vom A10-RTK Sensor und Ihrem Computer übertragen oder bearbeiten.

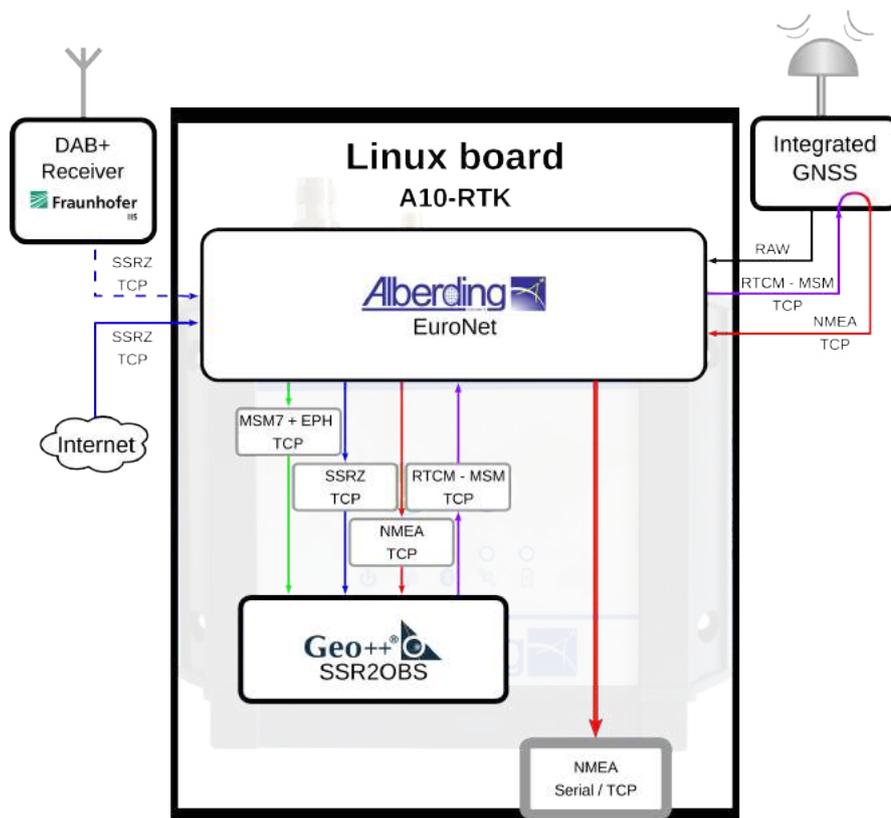


Wechseln Sie zu dem Pfad „/etc/euroнав/“, um die euronet.cfg Datei zu bearbeiten.

Beachten Sie, dass je nach Betriebssystem (Linux, Windows, etc.) Zeilenumbrüche in unterschiedlichen Formaten abgespeichert werden können (CR, LF). Die Darstellung der Textdokumente kann entsprechend falsch sein. Neben einigen anderen Editoren erkennen die Programme TextPad und Notepad++ die unterschiedlichen Zeilenumbrüche automatisch.

9 SSRZ mit dem Alberding A10-RTK

Mit den folgenden Konfigurationen des GNSS-Boards, der EURONET-Software und dem PPP-RTK Format der Firma GEO++[®] (ssr2obs.cfg) wird auf dem Alberding A10-RTK ein Tool für das Umwandeln der SSRZ-Daten in RTCM3-RTK bereitgestellt.



Die dargestellten Kommandos und Eingaben müssen als Beispiel angesehen werden. Bitte passen Sie die Skripte für die eigenen Anforderungen an.

Konfiguration GNSS-Board (am Beispiel des Septentrio Mosaic)



```
COM1: 115200 8N1 - NMEA GGA+RMC 1 sec
IPS1: Port 8888 TCP (send only) - NMEA GGA+RMC 1 sec
IPS2: Port 1804 TCP (send only) - RTCMv3 1019, 1020, 1042, 1044, 1046
IPS3: Port 8889 TCP2Way (send and receive) - RTCMv3 input
```

Konfiguration der Software EUONET (euronet.cfg)

Hinweis: Die Zeilen beginnend mit einer # werden als Kommentare angesehen und werden nicht ausgeführt.

```
--logdir = /var/euronav/log
--stderr = filea:/var/euronav/log/euronet.err
--stdout = filea:/var/euronav/log/euronet.log
--keyfile = /etc/euronav/euronet.key

# Serieller Port I/O
--socket = serial:NMEA_SER/38400@/dev/ttyS4%NMEA
--markername NMEA_SER
  # zum Testen
  --rawoutput = tcplist::1901
--endmarkername

# NMEA-Daten vom internen Mosaic
--socket = tcp:MOSAIC@10.14.75.2:8888%NMEA
--markername MOSAIC
  # zum Testen
  --rawoutput = tcplist::1902
  # NMEA zum SSR
  --rawoutput = tcplist:127.0.0.1:22002
  # zur seriellen Schnittstelle zum Test
  #--rawoutput = socket:NMEA_SER
--endmarkername

# RTCM3 Ephemeris vom Mosaic
--socket = tcp:INPUT_RAW_EPH@10.14.75.2:1802%RTCM3
--rtcmmarker OUTPUT_RTCM3
--markername OUTPUT_RTCM3
  # zum Testen
  --rtcmoutput = tcplist::1903
  # Ephemeriden für SSR2OBS
  --rtcmoutput = tcplist::22001
  --rtcmtype = 1019(10),1020(10),1042(10),1044(10),1046(10)
--endmarkername

--socket = tcp:SSROBS@127.0.0.1:22003%RTCM3
```



```
--markername SSROBS
# zum Testen
--rtcmdirect = tcplist::1905
# Korrekturdaten für Mosaic
--rtcmdirect = tcp:10.14.75.2:8889
--rtcmdirect = socket:NMEA_SER
--endmarkername

--socket = tcp:INPUT_SSR@192.168.200.10:1580%RTCM3
#--socket = ntrip:INPUT_SSR/user:password@ntrip.gnssonline.eu%RTCM3
--markername = INPUT_SSR
# zum Testen
--rtcmdirect = tcplist::1904
# to ssr2obs
--rtcmdirect = tcplist::22000
# to serial port
#--rtcmdirect = socket:NMEA_SER
--endmarkername
```

Konfigurationsdatei ssr2obs.cfg

Hinweis: Die Zeilen beginnend mit einer # werden als Kommentare angesehen und werden nicht ausgeführt.

```
# config file for /usr/lib/systemd/system/ssr2obs.service
# Check ssr2obs -? for more information

# for debug
# ssr2obs -z=127.0.0.1:22000 -eph=127.0.0.1:22001 -i=127.0.0.1:22002
  -oss=22003 +syst -rupd=-1.0 -rdist=100 -glo=SEPO -rtcmmsm -ri=1.0
  -ro=0.8 -re=1.0 -datumdef=/etc/euronav/etrfitrf.dat -ssrdatum=ITRF14
  -refdatum=ETRF_R2016 -debug=5121 &2>&1 |less

# SSRZ input stream
# activate SSRZ mode and connect to SSRZ server address [tcp_addr:port]
SSR_INPUT="-z=127.0.0.1:22000"

# Ephemeris input
# read ephemerides messages from server address [tcp_addr:port]
EPHEM_INPUT="-eph=127.0.0.1:22001"

# Position and time from Rover
# -i=address - rover pos and time from GGA+ZDA input
IPS_IO="-i=127.0.0.1:22002 -oss=22003"
```



```
# System time
# assume system time beeing correct to a few minutes (then NMEA ZDA not
required)
USE_SYSTEM_TIME="+syst"

# Virtual reference options
# -rupd - update virtual reference position (1.0 default value)
#       -1, the first GGa position is used
# -glo  - introduce GLONASS observation biases for GLONASS
#       UBX or SEP0, see -glo=?
# -rdist - distance of virtual reference from user position [100m]
VREF_OPTS="-rupd=-1.0 -rdist=100 -glo=SEP0"

# RTCM MSM
# enable RTCM MSM output (with SSRZ input only)
RTCM_OPTS="-rtcmmsm"

# Offsets
# -ri[=]x.x - output interval [1.0]
# -ro[=]x.x - output offset to last GGA [0.8]
#             was set to 0.2 from geo++ this
#             produced a data delay of -0.6 sec,
#             it is unusable for Javad Triumph from DLR
# -re[=]x   - output epoch offset to last GGA [1.0]
RATE_OPTS="-ri=1.0 -ro=0.8 -re=1.0"

# Transformation from ITRF08 (PPP) to ETRF_D (SAPOS)
# -datumdef=FNAME - datum trafo defined in file FNAME
# -ssrdatum=NAME  - assume SSR is in datum NAME
# -refdatum=NAME  - datum for DGP/RTCM output
# GEO_DATUM_OPTS="-datumdef=/etc/euronav/etrfitrf.dat -ssrdatum=ITRF14
-refdatum=ETRS89"
GEO_DATUM_OPTS="-datumdef=/etc/euronav/etrfitrf.dat -ssrdatum=ITRF14
-refdatum=ETRF_R2016"
# GEO_DATUM_OPTS=""
# debug options
DEBUG_OPTS=""
# DEBUG_OPTS="-debug=5121"
```



10 Haftungsausschluss

Die Alberding GmbH hat mit dem *A10-RTK* Sensor ein innovatives System für vielfältige Anwendungen der präzisen Positionierung entwickelt. Der *A10-RTK* Sensor vereint Komponenten unterschiedlicher Hersteller in einem Gehäuse. Die Sensoren wurden seitens der Alberding GmbH auf deren Eignung überprüft. Es bleibt jedoch ein Restrisiko, dass entweder einzelne Komponenten oder das System in der Zusammensetzung nicht korrekt arbeitet.

Die Alberding GmbH übernimmt weder die Haftung noch die Verantwortung für direkte und indirekte Kosten, die durch die Verwendung der Alberding Telemetrie- und Positionierungssysteme *A07-RTK* und *A10-RTK* entstehen. Der Hersteller behält sich das Recht vor, ohne vorherige Mitteilung, Änderungen bezüglich des Produkts, der technischen Daten oder der Bedienungsanleitung vorzunehmen.