

Inhaltsverzeichnis

1. DXL - APRStracker	23
2. Benutzer:OE2WAO	9
3. DXL - APRSmap	16
4. Datei:AprsTracker.zip	30
5. Datei:DXLtrackerConfig.zip	37
6. Datei:Dxl-aprsTracker-tiny13-10MHz-4800Bd.zip	44
7. Datei:Dxl-aprsTracker-tiny13-10MHz-9600Bd.zip	51
8. TCE Tinycore Linux Projekt	58

DXL - APRStracker

Versionsgeschichte interaktiv durchsuchen
VisuellWikitext

Version vom 24. September 2023, 20:11

Uhr (Quelltext anzeigen)

OE2WAO (Diskussion | Beiträge)

K

Markierung: 2017-Quelltext-Bearbeitung

← Zum vorherigen Versionsunterschied

Version vom 24. September 2023, 20:13

Uhr (Quelltext anzeigen)

OE2WAO (Diskussion | Beiträge)

K

Markierung: 2017-Quelltext-Bearbeitung

Zum nächsten Versionsunterschied →

Zeile 32:

|4800

|1200

–

[[Datei:**dxl**-aprsTracker-tiny13-10MHz-4800Bd.zip|verweis=Special:FilePath/dxl-aprsTracker-tiny13-10MHz-4800Bd.zip]]

|-

|ATMEL Tiny13

Zeile 32:

|4800

|1200

+

[[Datei:**Dxl**-aprsTracker-tiny13-10MHz-4800Bd.zip|verweis=Special:FilePath/dxl-aprsTracker-tiny13-10MHz-4800Bd.zip]]

|-

|ATMEL Tiny13

Zeile 38:

|9600

|1200

–

[[Datei:**dxl**-aprsTracker-tiny13-10MHz-9600Bd.zip|verweis=Special:FilePath/dxl-aprsTracker-tiny13-10MHz-9600Bd.zip]]

|-

|}

Zeile 38:

|9600

|1200

+

[[Datei:**Dxl**-aprsTracker-tiny13-10MHz-9600Bd.zip|verweis=Special:FilePath/dxl-aprsTracker-tiny13-10MHz-9600Bd.zip]]

|-

|}

Zeile 122:

[[Datei:Oe5eep_2.jpg|rechts|mini|hochkant|200px|Gesamtansicht|verweis=Special:FilePath/oe5eep_2.jpg]]

–

[[Datei:**oe5eep_1**.jpg|rechts|mini|200px|Gesamtansicht|verweis=Special:FilePath/oe5eep_1.jpg]]

Zeile 122:

[[Datei:Oe5eep_2.jpg|rechts|mini|hochkant|200px|Gesamtansicht|verweis=Special:FilePath/oe5eep_2.jpg]]

+

[[Datei:**Oe5eep_1**.jpg|rechts|mini|200px|Gesamtansicht|verweis=Special:FilePath/oe5eep_1.jpg]]

-	[[Datei:oe5eep_3.jpg rechts mini 200px Innenleben verweis=Special:FilePath/oe5eep_3.jpg]]	+	[[Datei:Oe5eep_3.jpg rechts mini 200px Innenleben verweis=Special:FilePath/oe5eep_3.jpg]]
-	[[Datei:oe5eep_4h.jpg rechts mini 200px Innenleben-Detail-Tracker+GPS-RX verweis=Special:FilePath/oe5eep_4h.jpg]]	+	[[Datei:Oe5eep_4h.jpg rechts mini 200px Innenleben-Detail-Tracker+GPS-RX verweis=Special:FilePath/oe5eep_4h.jpg]]

Version vom 24. September 2023, 20:13 Uhr

Inhaltsverzeichnis

1	Einleitung	25
2	Software	25
3	Hardware	26
3.1	Trackerschaltung	26
3.2	Geeignete GPS-Empfänger	26
4	Konfiguration	26
5	Referenzprojekte	27
5.1	OE5EEP, Heinz	27
6	Prämiert	28
7	LoRa Variante	29
8	Weitere Projekte	29

Einleitung

Der APRStracker von OE5DXL erlaubt es mit minimalstem Hardwareaufwand in der Betriebsart APRS QRV zu werden. Außerdem wurde bei der Entwicklung der Software darauf geachtet, die derzeitigen Möglichkeiten des APRS-Protokolls in Form von Mic-e optimalst auszunutzen. In der Praxis ist dies durch extrem kurze Frames erkennbar, was im Mobilbetrieb (QSB) erhebliche Vorteile bringt. Zu dem wird das schon etwas in die Jahre gekommene aber durchaus am effektivsten SSID-Pathrouting (im Configtool 'COMPRESSED' genannt) unterstützt.

Als Minimum an Hardware wird benötigt

- ein Mikrocontroller ATtiny13, ATtiny2313, Atmega88 oder auch andere Typen mit (Quarz)-Takt durch Anpassen der I/O Pins im Source
- Quarz frei wählbar ca. 6..20MHz (Im Source eingeben).
- PTT-Transistor
- RC-Tiefpass zum wegfiltern der PWM-Frequenz benötigt.

Software

Opensource Software von OE5DXL, in Assembler geschrieben, welche den Tracker zum Leben erweckt:

Datei:[AprsTracker.zip](#)

Der Assemblercode wird im einfachsten Fall mit dem Compiler 'gavrasm' [Gerd's AVR Assembler](#), welcher sowohl für Linux als auch für Windows verfügbar ist, kompiliert und anschließend in den µC gebrannt (z.B.: mit AVRdude [\[1\]](#) und dem USBasp Programmer [\[2\]](#)). Es ist auch möglich mit dem von ATMEL angebotenen AVR-Studio den Code zu übersetzen.

Fertige Kompilate:

Prozessor	Systemtakt	GPS-Baud	AFSK Baud	Download
ATMEL Tiny13	10 Mhz	4800	1200	Datei:Dxl-aprsTracker-tiny13-10MHz-4800Bd.zip
ATMEL Tiny13	10 Mhz	9600	1200	Datei:Dxl-aprsTracker-tiny13-10MHz-9600Bd.zip

Als Input dienen serielle GPS-Daten (GPRMC und GPGLL), der Tracker generiert daraus anhand der programmierten Konfiguration als Output (nebst PTT) die AFSK-Modulation im APRS-Mic-e Format mit Position, Geschwindigkeit, Fahrtrichtung und Höhe, welche dem Funkgerät zugeführt wird.

Dabei sind (pro Profil) 2 Bakenzeiten und eine Geschwindigkeit einstellbar unter/ueber der langsam/schnell gebakt wird.

Trackerschaltung

The diagram illustrates the internal circuitry of the APRS-Tracker module. Key components include:

- Microcontroller:** ATtiny13 (IC404) configured for 1MHz operation.
- Power Regulation:** A 5V regulator (VR401) and a 10MHz oscillator (Q401) provide stable power and timing.
- Serial Interface:** RS485 transceiver (IC401) for communication with the USB-to-UART bridge.
- Temperature Sensing:** A temperature sensor (T401) provides real-time temperature data.
- Connectors:** Headers for power (VIN, GND), data (VOUT, ENABLE), and temperature (TMOD, TPT, TGPS, TEND).

The ATtiny13 pinout is as follows:

Pin	Function
1	VIN
2	GND
3	VOUT
4	ENABLE
5	GND
6	OSC
7	OSC
8	VCC
9	RES
10	PB3
11	PB2
12	PB1
13	PB0
14	GND

Geeignete GPS-Empfänger

Hersteller	Type	Baudrate	Versorgungsspannung	Stromaufnahme	Preis	
Fastrax	UP501	9600	3.3V	~30mA	ca. 22€ (inkl. Versand)	http://clok.com/Seiten/Hersteller/Fastrax/UP501.pdf
Ublox	NEO-6M	9600	3.3V	~22mA	ca. 10€ (inkl. Versand)	Ublox
Globaltop	GTPA013	9600	3.3V	~20mA	ca. 19€	http://www.globaltop.com/da/Produkte/FGL/GTPA013V01

Konfiguration

Baudraten vom GPS bzw. PC zur Konfiguration sind je nach Prozessortakt im Bereich 300..200000, ueblich 4800, 9600.

AFSK-Baud und Tonfrequenzen (Shift) sind ebenso frei Konfigurierbar, gebräuchlich sind:

- 300 auf Kurzwelle
- 1200 auf UKW

Zur Einstellung von HUB bzw. Mikrofonpegel eignet sich am besten ein Poti.

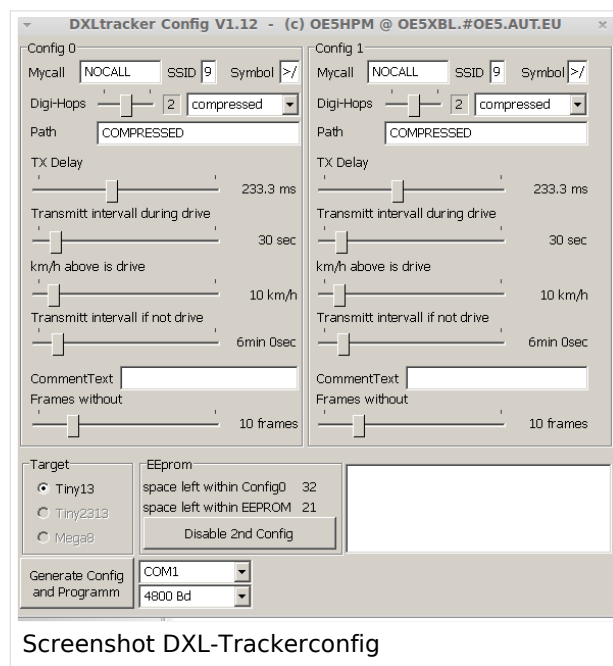
Um die PTT von Handfunkgeräten aufzutasten muss der Mikrofoneingang mit dem PTT Transistor kombiniert werden. Einfachste Variante ist im obigen Schaltungsbild sichtbar, der FET T402 zieht über einen 2k2 Widerstand den MIC-Eingang vom Handfunk gegen Masse und aktiviert dadurch die PTT, über C403 (100nF) wird die Modulation "eingekoppelt".

Als Option kann mit einem Jumper/Schalter (im Schaltbild SW401) zwischen 2 (am Tiny13) oder 4 (auf größeren Prozessoren) User-Profilen ausgewählt werden, zB. Fahrrad/Auto.

An einer Blink-Led für korrekten GPS-Empfang am Prozessor Pin wird gearbeitet.

Zur Konfiguration kommt ein kleines Tool von OE5HPM - DXLtrackerConfig zum Einsatz, dies generiert einen Konfigurationsstring für den DXLtracker und schickt selbigen über die serielle Schnittstelle raus.

Datei: [DXLtrackerConfig.zip](#)



Screenshot DXL-Trackerconfig

Referenzprojekte

OE5EEP, Heinz

Mein Aufbau eines DXL Modems mit einem Fastrax UP501 GPS Empfänger, den ich für Bergtouren im Rahmen des SOTA-Programms einsetzen möchte: Das DXL Modem wurde mir freundlicherweise vom ADL501 zur Verfügung gestellt, der UP501 ist über e-Bay zu beziehen. Als Spannungsquelle für beide Baugruppen gemeinsam dienen 4 NiMH AAA-Zellen von Conrad mit einer nominalen Kapazität von 1100mAh, was für einen ganzen Bergtag leicht ausreicht. Die stabilisierte Spannung von 3,3V für des GPS Modul kann nach Brückung einer Diode an Pin 7 des Sub-D Anschlusses des DXL Trackers abgegriffen werden.

Da ich der Verlässlichkeit von freien Verkabelungen nicht traue, hab ich das GPS Modul mittels der mitgelieferten Steckerleiste auf einer Lochrasterplatte aufgelötet und die Verkabelungen zum DXL Modem auf beiden Seiten der Lochrasterplatte auf eine 9-polige Sub-D Buchse geführt. Dazu kommt noch ein Aus/Ein Schalter und ein Halter für die NiMH Zellen. Eingebaut hab ich das Ganze in eine ehemalige Präsentationsverpackung von 2 Kugelschreibern. Dieses transparente Kunststoffgehäuse erlaubt es, das Aufleuchten der LED im Sendefall von außen zu erkennen und

passt von der Größe her gut zum verwendeten Handfunkgerät (siehe Fotos). Der Aufbau wird am Handfunkgerät einfach mit Gummibändern (Stücke von einem Fahrradschlauch) befestigt. Die einzige elektrische Verbindung zum Handfunkgerät ist eine Audioleitung zur Mikrofonbuchse, in meinem Fall mit einem 2,5mm Mono-Klinkenstecker.

Bisher hab ich den Tracker auf mehreren Bergtouren eingesetzt und positive Erfahrungen gemacht. Der GPS Empfänger findet binnen weniger Minuten eine Position. Die Empfindlichkeit ist ausgezeichnet. Teilweise kann ich im Inneren von Gebäuden eine Position bekommen, allerdings dann mit größerem Fehler. Jedenfalls ist der GPS-Empfänger deutlich empfindlicher als der in meinem LG Smartphone. Im Zusammenspiel mit einem schon etwas älteren IC-2e mit etwa 3W Sendeleistung und Gummiantenne ist es mir auf Touren in den Kalkalpen immer gelungen APRS-Meldungen im Gipfelbereich abzusetzen. Im Funkschatten von Bergen verlief das nicht so verlässlich. Das ist aber kein Problem der Kombination Tracker und Handfunkgerät sondern ist auf den dünnen Ausbau der sonstige APRS Infrastruktur zurückzuführen.

Dieses Selbstbauprojekt ist sowohl technisch als auch finanziell überschaubar. Ich hoffe, dass genaue Positionsmeldungen während meiner Bergtouren die SOTA Jäger unterstützen und gegebenenfalls zu meiner Sicherheit in den Bergen beitragen! Danke an OE5DXL für die Entwicklung der Basis für dieses nette Selbstbauprojekt!

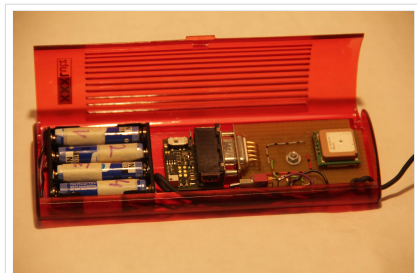
73 Heinz, OE5EEP



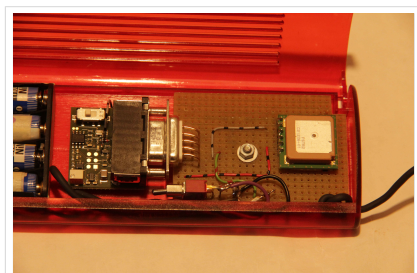
Gesamtansicht



Gesamtansicht



Innenleben



Innenleben-Detail-Tracker+GPS-RX

Prämiert

Der DXL APRStracker wurde mit dem € 2000,- dotierten ÖVSV Innovationspreis 2013 prämiert !
GRATULATION !!

LoRa Variante

Von OE5DXL gibt es auch eine LoRa Variante für Hardware-Tracker des Typs TTGO bzw. LILYGO inkl. Unterstützung für komprimierte Aussendung und Verwendung von MIC-E.

Zu finden ist die fertige INO-Datei in verschiedenen Ausführung, neben anderer Software, wie immer im [HAMNET Indexserver von OE5DXL](#).

Weitere Projekte

Weiter passende Projekte von OE5DXL aus dieser Serie sind

- [DXL - APRSmap](#) - Der neue APRS Client für Windows, Linux und ARM auf OSM Basis
- [TCE Tinycore Linux Projekt](#) - Der mächtige und innovative Digipeater für APRS, Packet Radio, Echolink, u.v.m.

DXL - APRStracker: Unterschied zwischen den Versionen

Versionsgeschichte interaktiv durchsuchen

VisuellWikitext

Version vom 24. September 2023, 20:11

Uhr (Quelltext anzeigen)

OE2WAO ([Diskussion](#) | [Beiträge](#))

K

Markierung: 2017-Quelltext-Bearbeitung

← Zum vorherigen Versionsunterschied

Version vom 24. September 2023, 20:13

Uhr (Quelltext anzeigen)

OE2WAO ([Diskussion](#) | [Beiträge](#))

K

Markierung: 2017-Quelltext-Bearbeitung

Zum nächsten Versionsunterschied →

Zeile 32:

|4800

|1200

– [[Datei:**dxl**-aprsTracker-tiny13-10MHz-4800Bd.zip|verweis=Special:FilePath/dxl-aprsTracker-tiny13-10MHz-4800Bd.zip]]

|-

|ATMEL Tiny13

Zeile 32:

|4800

|1200

+ [[Datei:**Dxl**-aprsTracker-tiny13-10MHz-4800Bd.zip|verweis=Special:FilePath/dxl-aprsTracker-tiny13-10MHz-4800Bd.zip]]

|-

|ATMEL Tiny13

Zeile 38:

|9600

|1200

– [[Datei:**dxl**-aprsTracker-tiny13-10MHz-9600Bd.zip|verweis=Special:FilePath/dxl-aprsTracker-tiny13-10MHz-9600Bd.zip]]

|-

|}

Zeile 38:

|9600

|1200

+ [[Datei:**Dxl**-aprsTracker-tiny13-10MHz-9600Bd.zip|verweis=Special:FilePath/dxl-aprsTracker-tiny13-10MHz-9600Bd.zip]]

|-

|}

Zeile 122:

[[Datei:Oe5eep_2.jpg|rechts|mini|hochkant|200px|Gesamtansicht|verweis=Special:FilePath/oe5eep_2.jpg]]

– [[Datei:**oe5eep_1**.jpg|rechts|mini|200px|Gesamtansicht|verweis=Special:FilePath/oe5eep_1.jpg]]

Zeile 122:

[[Datei:Oe5eep_2.jpg|rechts|mini|hochkant|200px|Gesamtansicht|verweis=Special:FilePath/oe5eep_2.jpg]]

+ [[Datei:**Oe5eep_1**.jpg|rechts|mini|200px|Gesamtansicht|verweis=Special:FilePath/oe5eep_1.jpg]]

-	[[Datei:oe5eep_3.jpg rechts mini 200px Innenleben verweis=Special:FilePath/oe5eep_3.jpg]]	+	[[Datei:Oe5eep_3.jpg rechts mini 200px Innenleben verweis=Special:FilePath/oe5eep_3.jpg]]
-	[[Datei:oe5eep_4h.jpg rechts mini 200px Innenleben-Detail-Tracker+GPS-RX verweis=Special:FilePath/oe5eep_4h.jpg]]	+	[[Datei:Oe5eep_4h.jpg rechts mini 200px Innenleben-Detail-Tracker+GPS-RX verweis=Special:FilePath/oe5eep_4h.jpg]]

Version vom 24. September 2023, 20:13 Uhr

Inhaltsverzeichnis

1	Einleitung	11
2	Software	11
3	Hardware	12
3.1	Trackerschaltung	12
3.2	Geeignete GPS-Empfänger	12
4	Konfiguration	12
5	Referenzprojekte	13
5.1	OE5EEP, Heinz	13
6	Prämiert	14
7	LoRa Variante	15
8	Weitere Projekte	15

Einleitung

Der APRStracker von OE5DXL erlaubt es mit minimalstem Hardwareaufwand in der Betriebsart APRS QRV zu werden. Außerdem wurde bei der Entwicklung der Software darauf geachtet, die derzeitigen Möglichkeiten des APRS-Protokolls in Form von Mic-e optimalst auszunutzen. In der Praxis ist dies durch extrem kurze Frames erkennbar, was im Mobilbetrieb (QSB) erhebliche Vorteile bringt. Zu dem wird das schon etwas in die Jahre gekommene aber durchaus am effektivsten SSID-Pathrouting (im Configtool 'COMPRESSED' genannt) unterstützt.

Als Minimum an Hardware wird benötigt

- ein Mikrocontroller ATtiny13, ATtiny2313, Atmega88 oder auch andere Typen mit (Quarz)-Takt durch Anpassen der I/O Pins im Source
- Quarz frei wählbar ca. 6..20MHz (Im Source eingeben).
- PTT-Transistor
- RC-Tiefpass zum wegfiltern der PWM-Frequenz benötigt.

Software

Opensource Software von OE5DXL, in Assembler geschrieben, welche den Tracker zum Leben erweckt:

Datei:[AprsTracker.zip](#)

Der Assemblercode wird im einfachsten Fall mit dem Compiler 'gavrasm' [Gerd's AVR Assembler](#), welcher sowohl für Linux als auch für Windows verfügbar ist, kompiliert und anschließend in den µC gebrannt (z.B.: mit AVRdude [\[1\]](#) und dem USBasp Programmer [\[2\]](#)). Es ist auch möglich mit dem von ATMEL angebotenen AVR-Studio den Code zu übersetzen.

Fertige Kompilate:

Prozessor	Systemtakt	GPS-Baud	AFSK Baud	Download
ATMEL Tiny13	10 Mhz	4800	1200	Datei:Dxl-aprsTracker-tiny13-10MHz-4800Bd.zip
ATMEL Tiny13	10 Mhz	9600	1200	Datei:Dxl-aprsTracker-tiny13-10MHz-9600Bd.zip

Als Input dienen serielle GPS-Daten (GPRMC und GPGLL), der Tracker generiert daraus anhand der programmierten Konfiguration als Output (nebst PTT) die AFSK-Modulation im APRS-Mic-e Format mit Position, Geschwindigkeit, Fahrtrichtung und Höhe, welche dem Funkgerät zugeführt wird.

Dabei sind (pro Profil) 2 Bakenzeiten und eine Geschwindigkeit einstellbar unter/ueber der langsam/schnell gebakt wird.

Trackerschaltung

The schematic diagram illustrates the APRS-Tracker circuit. It features a 5V supply (C405) and a 3.3V supply (C402). The circuit includes an NCP500SN33T1 voltage regulator, a 10MHz oscillator (Q401), a 2802A Schmitt trigger (S401), a PB5-nRES microcontroller (IC404), a 24SS13 timer (IC403), an AD8601ART op-amp (IC401), and a 24SS13 timer (IC403). It also features various passive components like resistors (R401-R409), capacitors (C401-C409), and diodes (D401, D404, D405, D406, D408, D409). The circuit is labeled 'APRS-Tracker'.

Geeignete GPS-Empfänger

Hersteller	Type	Baudrate	Versorgungsspannung	Stromaufnahme	Preis	
Fastrax	UP501	9600	3.3V	~30mA	ca. 22€ (inkl. Versand)	http://clok.com/Seiten/Hersteller/Fastrax/UP501.pdf
Ublox	NEO-6M	9600	3.3V	~22mA	ca. 10€ (inkl. Versand)	Ublox
Globaltop	GTPA013	9600	3.3V	~20mA	ca. 19€	http://www.globaltop.com/da/FGI/V0/

Konfiguration

Baudraten vom GPS bzw. PC zur Konfiguration sind je nach Prozessortakt im Bereich 300..200000, ueblich 4800, 9600.

AFSK-Baud und Tonfrequenzen (Shift) sind ebenso frei Konfigurierbar, gebräuchlich sind:

- 300 auf Kurzwelle
- 1200 auf UKW

Zur Einstellung von HUB bzw. Mikrofonpegel eignet sich am besten ein Poti.

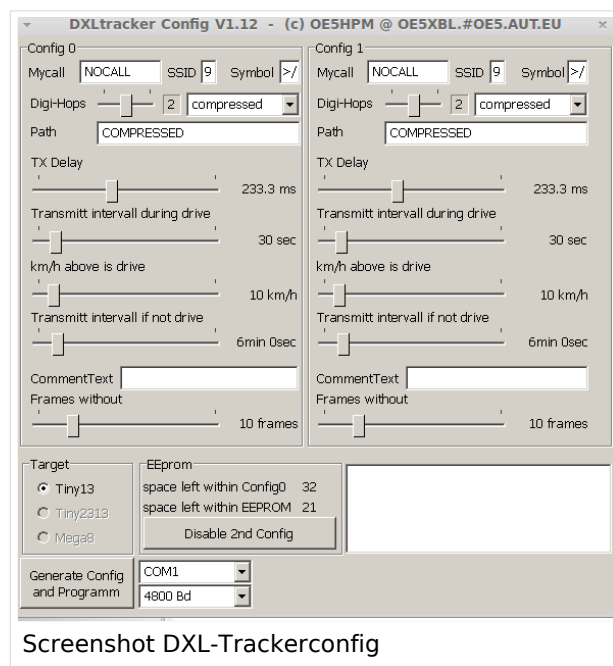
Um die PTT von Handfunkgeräten aufzutasten muss der Mikrofoneingang mit dem PTT Transistor kombiniert werden. Einfachste Variante ist im obigen Schaltungsbild sichtbar, der FET T402 zieht über einen 2k2 Widerstand den MIC-Eingang vom Handfunk gegen Masse und aktiviert dadurch die PTT, über C403 (100nF) wird die Modulation "eingekoppelt".

Als Option kann mit einem Jumper/Schalter (im Schaltbild SW401) zwischen 2 (am Tiny13) oder 4 (auf größeren Prozessoren) User-Profilen ausgewählt werden, zB. Fahrrad/Auto.

An einer Blink-Led für korrekten GPS-Empfang am Prozessor Pin wird gearbeitet.

Zur Konfiguration kommt ein kleines Tool von OE5HPM - DXLtrackerConfig zum Einsatz, dies generiert einen Konfigurationsstring für den DXLtracker und schickt selbigen über die serielle Schnittstelle raus.

[Datei:DXLtrackerConfig.zip](#)



Screenshot DXL-Trackerconfig

Referenzprojekte

OE5EEP, Heinz

Mein Aufbau eines DXL Modems mit einem Fastrax UP501 GPS Empfänger, den ich für Bergtouren im Rahmen des SOTA-Programms einsetzen möchte: Das DXL Modem wurde mir freundlicherweise vom ADL501 zur Verfügung gestellt, der UP501 ist über e-Bay zu beziehen. Als Spannungsquelle für beide Baugruppen gemeinsam dienen 4 NiMH AAA-Zellen von Conrad mit einer nominalen Kapazität von 1100mAh, was für einen ganzen Bergtag leicht ausreicht. Die stabilisierte Spannung von 3,3V für des GPS Modul kann nach Brückung einer Diode an Pin 7 des Sub-D Anschlusses des DXL Trackers abgegriffen werden.

Da ich der Verlässlichkeit von freien Verkabelungen nicht traue, hab ich das GPS Modul mittels der mitgelieferten Steckerleiste auf einer Lochrasterplatte aufgelötet und die Verkabelungen zum DXL Modem auf beiden Seiten der Lochrasterplatte auf eine 9-polige Sub-D Buchse geführt. Dazu kommt noch ein Aus/Ein Schalter und ein Halter für die NiMH Zellen. Eingebaut hab ich das Ganze in eine ehemalige Präsentationsverpackung von 2 Kugelschreibern. Dieses transparente Kunststoffgehäuse erlaubt es, das Aufleuchten der LED im Sendefall von außen zu erkennen und

passt von der Größe her gut zum verwendeten Handfunkgerät (siehe Fotos). Der Aufbau wird am Handfunkgerät einfach mit Gummibändern (Stücke von einem Fahrradschlauch) befestigt. Die einzige elektrische Verbindung zum Handfunkgerät ist eine Audioleitung zur Mikrofonbuchse, in meinem Fall mit einem 2,5mm Mono-Klinkenstecker.

Bisher hab ich den Tracker auf mehreren Bergtouren eingesetzt und positive Erfahrungen gemacht. Der GPS Empfänger findet binnen weniger Minuten eine Position. Die Empfindlichkeit ist ausgezeichnet. Teilweise kann ich im Inneren von Gebäuden eine Position bekommen, allerdings dann mit größerem Fehler. Jedenfalls ist der GPS-Empfänger deutlich empfindlicher als der in meinem LG Smartphone. Im Zusammenspiel mit einem schon etwas älteren IC-2e mit etwa 3W Sendeleistung und Gummiantenne ist es mir auf Touren in den Kalkalpen immer gelungen APRS-Meldungen im Gipfelbereich abzusetzen. Im Funkschatten von Bergen verlief das nicht so verlässlich. Das ist aber kein Problem der Kombination Tracker und Handfunkgerät sondern ist auf den dünnen Ausbau der sonstige APRS Infrastruktur zurückzuführen.

Dieses Selbstbauprojekt ist sowohl technisch als auch finanziell überschaubar. Ich hoffe, dass genaue Positionsmeldungen während meiner Bergtouren die SOTA Jäger unterstützen und gegebenenfalls zu meiner Sicherheit in den Bergen beitragen! Danke an OE5DXL für die Entwicklung der Basis für dieses nette Selbstbauprojekt!

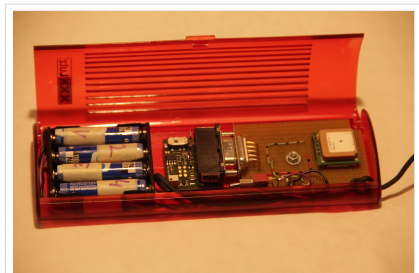
73 Heinz, OE5EEP



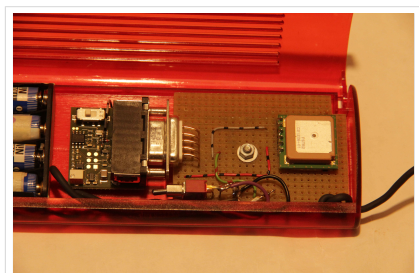
Gesamtansicht



Gesamtansicht



Innenleben



Innenleben-Detail-Tracker+GPS-RX

Prämiert

Der DXL APRStracker wurde mit dem € 2000,- dotierten ÖVSV Innovationspreis 2013 prämiert !
GRATULATION !!

LoRa Variante

Von OE5DXL gibt es auch eine LoRa Variante für Hardware-Tracker des Typs TTGO bzw. LILYGO inkl. Unterstützung für komprimierte Aussendung und Verwendung von MIC-E.

Zu finden ist die fertige INO-Datei in verschiedenen Ausführung, neben anderer Software, wie immer im [HAMNET Indexserver von OE5DXL](#).

Weitere Projekte

Weiter passende Projekte von OE5DXL aus dieser Serie sind

- [DXL - APRSmap](#) - Der neue APRS Client für Windows, Linux und ARM auf OSM Basis
- [TCE Tinycore Linux Projekt](#) - Der mächtige und innovative Digipeater für APRS, Packet Radio, Echolink, u.v.m.

DXL - APRStracker: Unterschied zwischen den Versionen

Versionsgeschichte interaktiv durchsuchen

VisuellWikitext

Version vom 24. September 2023, 20:11

Uhr (Quelltext anzeigen)

OE2WAO ([Diskussion](#) | [Beiträge](#))

K

Markierung: 2017-Quelltext-Bearbeitung

← Zum vorherigen Versionsunterschied

Version vom 24. September 2023, 20:13

Uhr (Quelltext anzeigen)

OE2WAO ([Diskussion](#) | [Beiträge](#))

K

Markierung: 2017-Quelltext-Bearbeitung

Zum nächsten Versionsunterschied →

Zeile 32:

|4800

|1200

– [[Datei:**dxl**-aprsTracker-tiny13-10MHz-4800Bd.zip|verweis=Special:FilePath/dxl-aprsTracker-tiny13-10MHz-4800Bd.zip]]

|-

|ATMEL Tiny13

Zeile 32:

|4800

|1200

+ [[Datei:**Dxl**-aprsTracker-tiny13-10MHz-4800Bd.zip|verweis=Special:FilePath/dxl-aprsTracker-tiny13-10MHz-4800Bd.zip]]

|-

|ATMEL Tiny13

Zeile 38:

|9600

|1200

– [[Datei:**dxl**-aprsTracker-tiny13-10MHz-9600Bd.zip|verweis=Special:FilePath/dxl-aprsTracker-tiny13-10MHz-9600Bd.zip]]

|-

|}

Zeile 38:

|9600

|1200

+ [[Datei:**Dxl**-aprsTracker-tiny13-10MHz-9600Bd.zip|verweis=Special:FilePath/dxl-aprsTracker-tiny13-10MHz-9600Bd.zip]]

|-

|}

Zeile 122:

[[Datei:Oe5eep_2.jpg|rechts|mini|hochkant|200px|Gesamtansicht|verweis=Special:FilePath/oe5eep_2.jpg]]

– [[Datei:**oe5eep_1**.jpg|rechts|mini|200px|Gesamtansicht|verweis=Special:FilePath/oe5eep_1.jpg]]

Zeile 122:

[[Datei:Oe5eep_2.jpg|rechts|mini|hochkant|200px|Gesamtansicht|verweis=Special:FilePath/oe5eep_2.jpg]]

+ [[Datei:**Oe5eep_1**.jpg|rechts|mini|200px|Gesamtansicht|verweis=Special:FilePath/oe5eep_1.jpg]]

-	[[Datei:oe5eep_3.jpg rechts mini 200px Innenleben verweis=Special:FilePath/oe5eep_3.jpg]]	+	[[Datei:Oe5eep_3.jpg rechts mini 200px Innenleben verweis=Special:FilePath/oe5eep_3.jpg]]
-	[[Datei:oe5eep_4h.jpg rechts mini 200px Innenleben-Detail-Tracker+GPS-RX verweis=Special:FilePath/oe5eep_4h.jpg]]	+	[[Datei:Oe5eep_4h.jpg rechts mini 200px Innenleben-Detail-Tracker+GPS-RX verweis=Special:FilePath/oe5eep_4h.jpg]]

Version vom 24. September 2023, 20:13 Uhr

Inhaltsverzeichnis

1	Einleitung	18
2	Software	18
3	Hardware	19
3.1	Trackerschaltung	19
3.2	Geeignete GPS-Empfänger	19
4	Konfiguration	19
5	Referenzprojekte	20
5.1	OE5EEP, Heinz	20
6	Prämiert	21
7	LoRa Variante	22
8	Weitere Projekte	22

Einleitung

Der APRStracker von OE5DXL erlaubt es mit minimalstem Hardwareaufwand in der Betriebsart APRS QRV zu werden. Außerdem wurde bei der Entwicklung der Software darauf geachtet, die derzeitigen Möglichkeiten des APRS-Protokolls in Form von Mic-e optimalst auszunutzen. In der Praxis ist dies durch extrem kurze Frames erkennbar, was im Mobilbetrieb (QSB) erhebliche Vorteile bringt. Zu dem wird das schon etwas in die Jahre gekommene aber durchaus am effektivsten SSID-Pathrouting (im Configtool 'COMPRESSED' genannt) unterstützt.

Als Minimum an Hardware wird benötigt

- ein Mikrocontroller ATtiny13, ATtiny2313, Atmega88 oder auch andere Typen mit (Quarz)-Takt durch Anpassen der I/O Pins im Source
- Quarz frei wählbar ca. 6..20MHz (Im Source eingeben).
- PTT-Transistor
- RC-Tiefpass zum wegfiltern der PWM-Frequenz benötigt.

Software

Opensource Software von OE5DXL, in Assembler geschrieben, welche den Tracker zum Leben erweckt:

Datei: [AprsTracker.zip](#)

Der Assemblercode wird im einfachsten Fall mit dem Compiler 'gavrasm' [Gerd's AVR Assembler](#), welcher sowohl für Linux als auch für Windows verfügbar ist, kompiliert und anschließend in den µC gebrannt (z.B.: mit AVRdude [\[1\]](#) und dem USBasp Programmer [\[2\]](#)). Es ist auch möglich mit dem von ATMEL angebotenen AVR-Studio den Code zu übersetzen.

Fertige Kompilate:

Prozessor	Systemtakt	GPS-Baud	AFSK Baud	Download
ATMEL Tiny13	10 Mhz	4800	1200	Datei:Dxl-aprsTracker-tiny13-10MHz-4800Bd.zip
ATMEL Tiny13	10 Mhz	9600	1200	Datei:Dxl-aprsTracker-tiny13-10MHz-9600Bd.zip

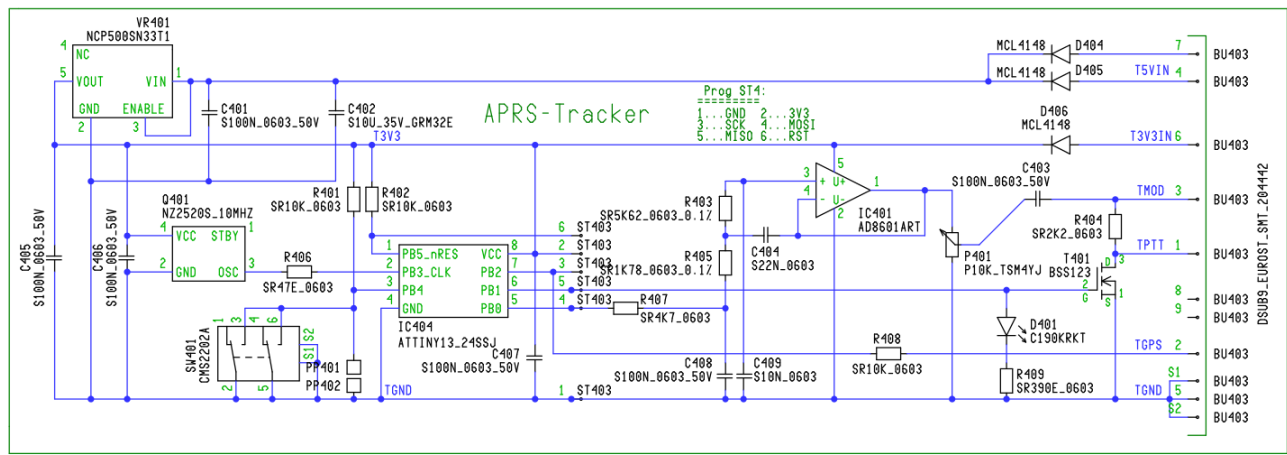
Als Input dienen serielle GPS-Daten (GPRMC und GPGLL), der Tracker generiert daraus anhand der programmierten Konfiguration als Output (nebst PTT) die AFSK-Modulation im APRS-Mic-e Format mit Position, Geschwindigkeit, Fahrtrichtung und Höhe, welche dem Funkgerät zugeführt wird.

Dabei sind (pro Profil) 2 Bakenzeiten und eine Geschwindigkeit einstellbar unter/ueber der langsam/schnell gebakt wird.

Hardware

Trackerschaltung

Schaltungsvorschlag von OE5HPM mit einem Tiny13 (kleinster Prozessor), wie er bereits mehrfach im Einsatz ist:



Detaillierte Schaltungsbeschreibung folgt.

Geeignete GPS-Empfänger

Hersteller	Type	Baudrate	Versorgungsspannung	Stromaufnahme	Preis	
Fastrax	UP501	9600	3.3V	~30mA	ca. 22€ (inkl. Versand)	http://clo.../Se...http://da.../UP...pdf
Ublox	NEO-6M	9600	3.3V	~22mA	ca. 10€ (inkl. Versand)	Ubl...
Globaltop	GTPA013	9600	3.3V	~20mA	ca. 19€	http://da...FGI...VO...

Konfiguration

GPS und Config-Programm liefern die seriellen Daten in TTL- oder RS232-Pegel. Der Pegel wird mittels einem (10k) Widerstand und der im Prozessor eingebauten Schutzdiode angepasst, die Polarisation erkennt die Software automatisch.

Baudraten vom GPS bzw. PC zur Konfiguration sind je nach Prozessortakt im Bereich 300..200000, ueblich 4800, 9600.

AFSK-Baud und Tonfrequenzen (Shift) sind ebenso frei Konfigurierbar, gebräuchlich sind:

- 300 auf Kurzwelle
- 1200 auf UKW

Zur Einstellung von HUB bzw. Mikrofonpegel eignet sich am besten ein Poti.

Um die PTT von Handfunkgeräten aufzutasten muss der Mikrofoneingang mit dem PTT Transistor kombiniert werden. Einfachste Variante ist im obigen Schaltungsbild sichtbar, der FET T402 zieht über einen 2k2 Widerstand den MIC-Eingang vom Handfunk gegen Masse und aktiviert dadurch die PTT, über C403 (100nF) wird die Modulation "eingekoppelt".

Als Option kann mit einem Jumper/Schalter (im Schaltbild SW401) zwischen 2 (am Tiny13) oder 4 (auf größeren Prozessoren) User-Profilen ausgewählt werden, zB. Fahrrad/Auto.

An einer Blink-Led für korrekten GPS-Empfang am Prozessor Pin wird gearbeitet.

Zur Konfiguration kommt ein kleines Tool von OE5HPM - DXLtrackerConfig zum Einsatz, dies generiert einen Konfigurationsstring für den DXLtracker und schickt selbigen über die serielle Schnittstelle raus.

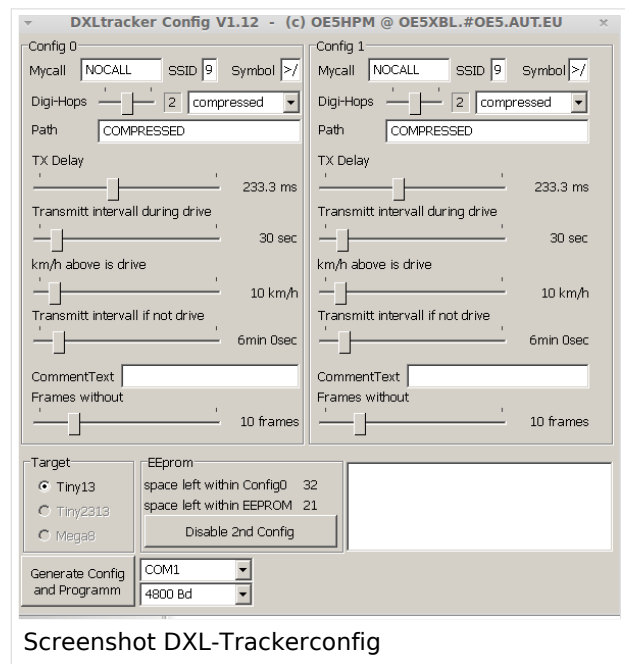
[Datei:DXLtrackerConfig.zip](#)

Referenzprojekte

OE5EEP, Heinz

Mein Aufbau eines DXL Modems mit einem Fastrax UP501 GPS Empfänger, den ich für Bergtouren im Rahmen des SOTA-Programms einsetzen möchte: Das DXL Modem wurde mir freundlicherweise vom ADL501 zur Verfügung gestellt, der UP501 ist über e-Bay zu beziehen. Als Spannungsquelle für beide Baugruppen gemeinsam dienen 4 NiMH AAA-Zellen von Conrad mit einer nominalen Kapazität von 1100mAh, was für einen ganzen Bergtag leicht ausreicht. Die stabilisierte Spannung von 3,3V für des GPS Modul kann nach Brückung einer Diode an Pin 7 des Sub-D Anschlusses des DXL Trackers abgegriffen werden.

Da ich der Verlässlichkeit von freien Verkabelungen nicht traue, hab ich das GPS Modul mittels der mitgelieferten Steckerleiste auf einer Lochrasterplatte aufgelötet und die Verkabelungen zum DXL Modem auf beiden Seiten der Lochrasterplatte auf eine 9-polige Sub-D Buchse geführt. Dazu kommt noch ein Aus/Ein Schalter und ein Halter für die NiMH Zellen. Eingebaut hab ich das Ganze in eine ehemalige Präsentationsverpackung von 2 Kugelschreibern. Dieses transparente Kunststoffgehäuse erlaubt es, das Aufleuchten der LED im Sendefall von außen zu erkennen und



Screenshot DXL-Trackerconfig

passt von der Größe her gut zum verwendeten Handfunkgerät (siehe Fotos). Der Aufbau wird am Handfunkgerät einfach mit Gummibändern (Stücke von einem Fahrradschlauch) befestigt. Die einzige elektrische Verbindung zum Handfunkgerät ist eine Audioleitung zur Mikrofonbuchse, in meinem Fall mit einem 2,5mm Mono-Klinkenstecker.

Bisher hab ich den Tracker auf mehreren Bergtouren eingesetzt und positive Erfahrungen gemacht. Der GPS Empfänger findet binnen weniger Minuten eine Position. Die Empfindlichkeit ist ausgezeichnet. Teilweise kann ich im Inneren von Gebäuden eine Position bekommen, allerdings dann mit größerem Fehler. Jedenfalls ist der GPS-Empfänger deutlich empfindlicher als der in meinem LG Smartphone. Im Zusammenspiel mit einem schon etwas älteren IC-2e mit etwa 3W Sendeleistung und Gummiantenne ist es mir auf Touren in den Kalkalpen immer gelungen APRS-Meldungen im Gipfelbereich abzusetzen. Im Funkschatten von Bergen verlief das nicht so verlässlich. Das ist aber kein Problem der Kombination Tracker und Handfunkgerät sondern ist auf den dünnen Ausbau der sonstige APRS Infrastruktur zurückzuführen.

Dieses Selbstbauprojekt ist sowohl technisch als auch finanziell überschaubar. Ich hoffe, dass genaue Positionsmeldungen während meiner Bergtouren die SOTA Jäger unterstützen und gegebenenfalls zu meiner Sicherheit in den Bergen beitragen! Danke an OE5DXL für die Entwicklung der Basis für dieses nette Selbstbauprojekt!

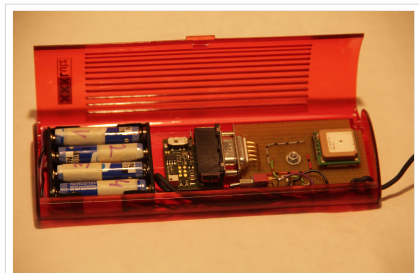
73 Heinz, OE5EEP



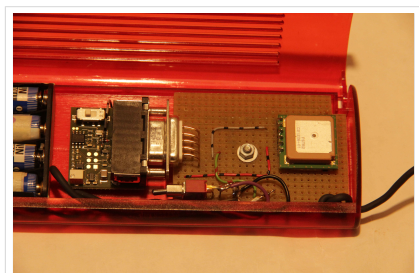
Gesamtansicht



Gesamtansicht



Innenleben



Innenleben-Detail-Tracker+GPS-RX

Prämiert

Der DXL APRStracker wurde mit dem € 2000,- dotierten ÖVSV Innovationspreis 2013 prämiert !
GRATULATION !!

LoRa Variante

Von OE5DXL gibt es auch eine LoRa Variante für Hardware-Tracker des Typs TTGO bzw. LILYGO inkl. Unterstützung für komprimierte Aussendung und Verwendung von MIC-E.

Zu finden ist die fertige INO-Datei in verschiedenen Ausführung, neben anderer Software, wie immer im [HAMNET Indexserver von OE5DXL](#).

Weitere Projekte

Weiter passende Projekte von OE5DXL aus dieser Serie sind

- [DXL - APRSmap](#) - Der neue APRS Client für Windows, Linux und ARM auf OSM Basis
- [TCE Tinycore Linux Projekt](#) - Der mächtige und innovative Digipeater für APRS, Packet Radio, Echolink, u.v.m.

DXL - APRTracker: Unterschied zwischen den Versionen

Versionsgeschichte interaktiv durchsuchen

VisuellWikitext

Version vom 24. September 2023, 20:11

Uhr (Quelltext anzeigen)

OE2WAO ([Diskussion](#) | [Beiträge](#))

K

Markierung: 2017-Quelltext-Bearbeitung

← Zum vorherigen Versionsunterschied

Version vom 24. September 2023, 20:13

Uhr (Quelltext anzeigen)

OE2WAO ([Diskussion](#) | [Beiträge](#))

K

Markierung: 2017-Quelltext-Bearbeitung

Zum nächsten Versionsunterschied →

Zeile 32:

|4800

|1200

– [[Datei:**dxl**-aprsTracker-tiny13-10MHz-4800Bd.zip|verweis=Special:FilePath/dxl-aprsTracker-tiny13-10MHz-4800Bd.zip]]

|-

|ATMEL Tiny13

Zeile 32:

|4800

|1200

+ [[Datei:**Dxl**-aprsTracker-tiny13-10MHz-4800Bd.zip|verweis=Special:FilePath/dxl-aprsTracker-tiny13-10MHz-4800Bd.zip]]

|-

|ATMEL Tiny13

Zeile 38:

|9600

|1200

– [[Datei:**dxl**-aprsTracker-tiny13-10MHz-9600Bd.zip|verweis=Special:FilePath/dxl-aprsTracker-tiny13-10MHz-9600Bd.zip]]

|-

|}

Zeile 38:

|9600

|1200

+ [[Datei:**Dxl**-aprsTracker-tiny13-10MHz-9600Bd.zip|verweis=Special:FilePath/dxl-aprsTracker-tiny13-10MHz-9600Bd.zip]]

|-

|}

Zeile 122:

[[Datei:Oe5eep_2.jpg|rechts|mini|hochkant|200px|Gesamtansicht|verweis=Special:FilePath/oe5eep_2.jpg]]

– [[Datei:**oe5eep_1**.jpg|rechts|mini|200px|Gesamtansicht|verweis=Special:FilePath/oe5eep_1.jpg]]

Zeile 122:

[[Datei:Oe5eep_2.jpg|rechts|mini|hochkant|200px|Gesamtansicht|verweis=Special:FilePath/oe5eep_2.jpg]]

+ [[Datei:**Oe5eep_1**.jpg|rechts|mini|200px|Gesamtansicht|verweis=Special:FilePath/oe5eep_1.jpg]]

-	[[Datei:oe5eep_3.jpg rechts mini 200px Innenleben verweis=Special:FilePath/oe5eep_3.jpg]]	+	[[Datei:Oe5eep_3.jpg rechts mini 200px Innenleben verweis=Special:FilePath/oe5eep_3.jpg]]
-	[[Datei:oe5eep_4h.jpg rechts mini 200px Innenleben-Detail-Tracker+GPS-RX verweis=Special:FilePath/oe5eep_4h.jpg]]	+	[[Datei:Oe5eep_4h.jpg rechts mini 200px Innenleben-Detail-Tracker+GPS-RX verweis=Special:FilePath/oe5eep_4h.jpg]]

Version vom 24. September 2023, 20:13 Uhr

Inhaltsverzeichnis

1	Einleitung	25
2	Software	25
3	Hardware	26
3.1	Trackerschaltung	26
3.2	Geeignete GPS-Empfänger	26
4	Konfiguration	26
5	Referenzprojekte	27
5.1	OE5EEP, Heinz	27
6	Prämiert	28
7	LoRa Variante	29
8	Weitere Projekte	29

Einleitung

Der APRStracker von OE5DXL erlaubt es mit minimalstem Hardwareaufwand in der Betriebsart APRS QRV zu werden. Außerdem wurde bei der Entwicklung der Software darauf geachtet, die derzeitigen Möglichkeiten des APRS-Protokolls in Form von Mic-e optimalst auszunutzen. In der Praxis ist dies durch extrem kurze Frames erkennbar, was im Mobilbetrieb (QSB) erhebliche Vorteile bringt. Zu dem wird das schon etwas in die Jahre gekommene aber durchaus am effektivsten SSID-Pathrouting (im Configtool 'COMPRESSED' genannt) unterstützt.

Als Minimum an Hardware wird benötigt

- ein Mikrocontroller ATtiny13, ATtiny2313, Atmega88 oder auch andere Typen mit (Quarz)-Takt durch Anpassen der I/O Pins im Source
- Quarz frei wählbar ca. 6..20MHz (Im Source eingeben).
- PTT-Transistor
- RC-Tiefpass zum wegfiltern der PWM-Frequenz benötigt.

Software

Opensource Software von OE5DXL, in Assembler geschrieben, welche den Tracker zum Leben erweckt:

Datei:[AprsTracker.zip](#)

Der Assemblercode wird im einfachsten Fall mit dem Compiler 'gavrasm' [Gerd's AVR Assembler](#), welcher sowohl für Linux als auch für Windows verfügbar ist, kompiliert und anschließend in den µC gebrannt (z.B.: mit AVRdude [\[1\]](#) und dem USBasp Programmer [\[2\]](#)). Es ist auch möglich mit dem von ATMEL angebotenen AVR-Studio den Code zu übersetzen.

Fertige Kompilate:

Prozessor	Systemtakt	GPS-Baud	AFSK Baud	Download
ATMEL Tiny13	10 Mhz	4800	1200	Datei:Dxl-aprsTracker-tiny13-10MHz-4800Bd.zip
ATMEL Tiny13	10 Mhz	9600	1200	Datei:Dxl-aprsTracker-tiny13-10MHz-9600Bd.zip

Als Input dienen serielle GPS-Daten (GPRMC und GPGLL), der Tracker generiert daraus anhand der programmierten Konfiguration als Output (nebst PTT) die AFSK-Modulation im APRS-Mic-e Format mit Position, Geschwindigkeit, Fahrtrichtung und Höhe, welche dem Funkgerät zugeführt wird.

Dabei sind (pro Profil) 2 Bakenzeiten und eine Geschwindigkeit einstellbar unter/ueber der langsam/schnell gebakt wird.

Trackerschaltung

[illegible]

Geeignete GPS-Empfänger

Hersteller	Type	Baudrate	Versorgungsspannung	Stromaufnahme	Preis	
Fastrax	UP501	9600	3.3V	~30mA	ca. 22€ (inkl. Versand)	http://clok.com/Seiten/FAQs/FAQs-UP501.pdf
Ublox	NEO-6M	9600	3.3V	~22mA	ca. 10€ (inkl. Versand)	Ublox
Globaltop	GTPA013	9600	3.3V	~20mA	ca. 19€	http://daikin.com/Files/GT-V01

Baudraten vom GPS bzw. PC zur Konfiguration sind je nach Prozessortakt im Bereich 300..200000, ueblich 4800, 9600.

AFSK-Baud und Tonfrequenzen (Shift) sind ebenso frei Konfigurierbar, gebräuchlich sind:

- 300 auf Kurzwelle
- 1200 auf UKW

Zur Einstellung von HUB bzw. Mikrofonpegel eignet sich am besten ein Poti.

Um die PTT von Handfunkgeräten aufzutasten muss der Mikrofoneingang mit dem PTT Transistor kombiniert werden. Einfachste Variante ist im obigen Schaltungsbild sichtbar, der FET T402 zieht über einen 2k2 Widerstand den MIC-Eingang vom Handfunk gegen Masse und aktiviert dadurch die PTT, über C403 (100nF) wird die Modulation "eingekoppelt".

Als Option kann mit einem Jumper/Schalter (im Schaltbild SW401) zwischen 2 (am Tiny13) oder 4 (auf größeren Prozessoren) User-Profilen ausgewählt werden, zB. Fahrrad/Auto.

An einer Blink-Led für korrekten GPS-Empfang am Prozessor Pin wird gearbeitet.

Zur Konfiguration kommt ein kleines Tool von OE5HPM - DXLtrackerConfig zum Einsatz, dies generiert einen Konfigurationsstring für den DXLtracker und schickt selbigen über die serielle Schnittstelle raus.

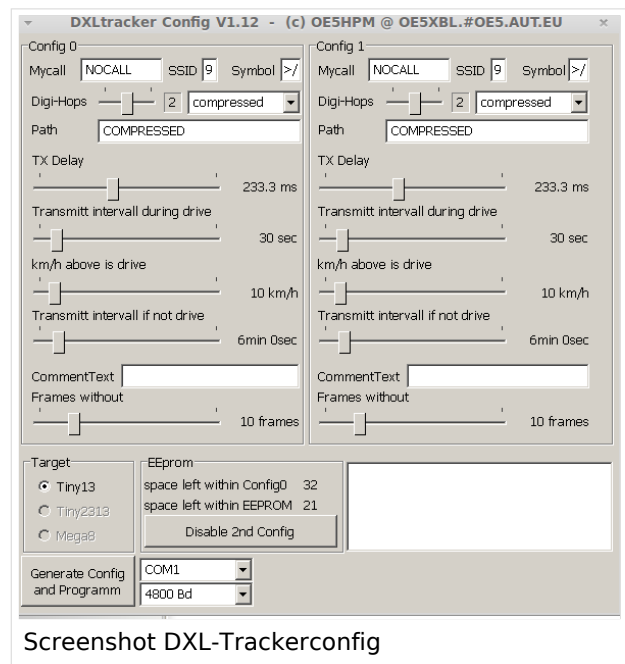
[Datei:DXLtrackerConfig.zip](#)

Referenzprojekte

OE5EEP, Heinz

Mein Aufbau eines DXL Modems mit einem Fastrax UP501 GPS Empfänger, den ich für Bergtouren im Rahmen des SOTA-Programms einsetzen möchte: Das DXL Modem wurde mir freundlicherweise vom ADL501 zur Verfügung gestellt, der UP501 ist über e-Bay zu beziehen. Als Spannungsquelle für beide Baugruppen gemeinsam dienen 4 NiMH AAA-Zellen von Conrad mit einer nominalen Kapazität von 1100mAh, was für einen ganzen Bergtag leicht ausreicht. Die stabilisierte Spannung von 3,3V für des GPS Modul kann nach Brückung einer Diode an Pin 7 des Sub-D Anschlusses des DXL Trackers abgegriffen werden.

Da ich der Verlässlichkeit von freien Verkabelungen nicht traue, hab ich das GPS Modul mittels der mitgelieferten Steckerleiste auf einer Lochrasterplatte aufgelötet und die Verkabelungen zum DXL Modem auf beiden Seiten der Lochrasterplatte auf eine 9-polige Sub-D Buchse geführt. Dazu kommt noch ein Aus/Ein Schalter und ein Halter für die NiMH Zellen. Eingebaut hab ich das Ganze in eine ehemalige Präsentationsverpackung von 2 Kugelschreibern. Dieses transparente Kunststoffgehäuse erlaubt es, das Aufleuchten der LED im Sendefall von außen zu erkennen und



Screenshot DXL-Trackerconfig

passt von der Größe her gut zum verwendeten Handfunkgerät (siehe Fotos). Der Aufbau wird am Handfunkgerät einfach mit Gummibändern (Stücke von einem Fahrradschlauch) befestigt. Die einzige elektrische Verbindung zum Handfunkgerät ist eine Audioleitung zur Mikrofonbuchse, in meinem Fall mit einem 2,5mm Mono-Klinkenstecker.

Bisher hab ich den Tracker auf mehreren Bergtouren eingesetzt und positive Erfahrungen gemacht. Der GPS Empfänger findet binnen weniger Minuten eine Position. Die Empfindlichkeit ist ausgezeichnet. Teilweise kann ich im Inneren von Gebäuden eine Position bekommen, allerdings dann mit größerem Fehler. Jedenfalls ist der GPS-Empfänger deutlich empfindlicher als der in meinem LG Smartphone. Im Zusammenspiel mit einem schon etwas älteren IC-2e mit etwa 3W Sendeleistung und Gummiantenne ist es mir auf Touren in den Kalkalpen immer gelungen APRS-Meldungen im Gipfelbereich abzusetzen. Im Funkschatten von Bergen verlief das nicht so verlässlich. Das ist aber kein Problem der Kombination Tracker und Handfunkgerät sondern ist auf den dünnen Ausbau der sonstige APRS Infrastruktur zurückzuführen.

Dieses Selbstbauprojekt ist sowohl technisch als auch finanziell überschaubar. Ich hoffe, dass genaue Positionsmeldungen während meiner Bergtouren die SOTA Jäger unterstützen und gegebenenfalls zu meiner Sicherheit in den Bergen beitragen! Danke an OE5DXL für die Entwicklung der Basis für dieses nette Selbstbauprojekt!

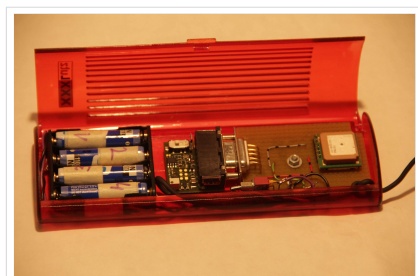
73 Heinz, OE5EEP



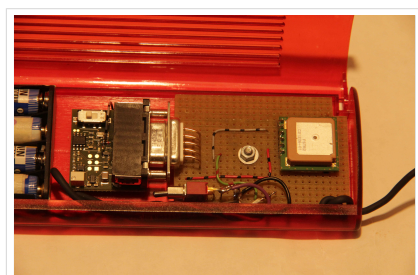
Gesamtansicht



Gesamtansicht



Innenleben



Innenleben-Detail-Tracker+GPS-RX

Prämiert

Der DXL APRStracker wurde mit dem € 2000,- dotierten ÖVSV Innovationspreis 2013 prämiert !
GRATULATION !!

LoRa Variante

Von OE5DXL gibt es auch eine LoRa Variante für Hardware-Tracker des Typs TTGO bzw. LILYGO inkl. Unterstützung für komprimierte Aussendung und Verwendung von MIC-E.

Zu finden ist die fertige INO-Datei in verschiedenen Ausführung, neben anderer Software, wie immer im [HAMNET Indexserver von OE5DXL](#).

Weitere Projekte

Weiter passende Projekte von OE5DXL aus dieser Serie sind

- [DXL - APRSmap](#) - Der neue APRS Client für Windows, Linux und ARM auf OSM Basis
- [TCE Tinycore Linux Projekt](#) - Der mächtige und innovative Digipeater für APRS, Packet Radio, Echolink, u.v.m.

DXL - APRTracker: Unterschied zwischen den Versionen

[Versionsgeschichte interaktiv durchsuchen](#)

[Visuell Wikitext](#)

Version vom 24. September 2023, 20:11

Uhr (Quelltext anzeigen)

[OE2WAO](#) ([Diskussion](#) | [Beiträge](#))

K

Markierung: 2017-Quelltext-Bearbeitung

[← Zum vorherigen Versionsunterschied](#)

Version vom 24. September 2023, 20:13

Uhr (Quelltext anzeigen)

[OE2WAO](#) ([Diskussion](#) | [Beiträge](#))

K

Markierung: 2017-Quelltext-Bearbeitung

[Zum nächsten Versionsunterschied →](#)

Zeile 32:

|4800

|1200

– [[Datei:**dxl**-aprsTracker-tiny13-10MHz-4800Bd.zip|verweis=Special:FilePath/dxl-aprsTracker-tiny13-10MHz-4800Bd.zip]]

|-

|ATMEL Tiny13

Zeile 32:

|4800

|1200

+ [[Datei:**Dxl**-aprsTracker-tiny13-10MHz-4800Bd.zip|verweis=Special:FilePath/dxl-aprsTracker-tiny13-10MHz-4800Bd.zip]]

|-

|ATMEL Tiny13

Zeile 38:

|9600

|1200

– [[Datei:**dxl**-aprsTracker-tiny13-10MHz-9600Bd.zip|verweis=Special:FilePath/dxl-aprsTracker-tiny13-10MHz-9600Bd.zip]]

|-

|}

Zeile 38:

|9600

|1200

+ [[Datei:**Dxl**-aprsTracker-tiny13-10MHz-9600Bd.zip|verweis=Special:FilePath/dxl-aprsTracker-tiny13-10MHz-9600Bd.zip]]

|-

|}

Zeile 122:

[[Datei:Oe5eep_2.jpg|rechts|mini|hochkant|200px|Gesamtansicht|verweis=Special:FilePath/oe5eep_2.jpg]]

– [[Datei:**oe5eep_1**.jpg|rechts|mini|200px|Gesamtansicht|verweis=Special:FilePath/oe5eep_1.jpg]]

Zeile 122:

[[Datei:Oe5eep_2.jpg|rechts|mini|hochkant|200px|Gesamtansicht|verweis=Special:FilePath/oe5eep_2.jpg]]

+ [[Datei:**Oe5eep_1**.jpg|rechts|mini|200px|Gesamtansicht|verweis=Special:FilePath/oe5eep_1.jpg]]

-	[[Datei:oe5eep_3.jpg rechts mini 200px Innenleben verweis=Special:FilePath/oe5eep_3.jpg]]	+	[[Datei:Oe5eep_3.jpg rechts mini 200px Innenleben verweis=Special:FilePath/oe5eep_3.jpg]]
-	[[Datei:oe5eep_4h.jpg rechts mini 200px Innenleben-Detail-Tracker+GPS-RX verweis=Special:FilePath/oe5eep_4h.jpg]]	+	[[Datei:Oe5eep_4h.jpg rechts mini 200px Innenleben-Detail-Tracker+GPS-RX verweis=Special:FilePath/oe5eep_4h.jpg]]

Version vom 24. September 2023, 20:13 Uhr

Inhaltsverzeichnis

1	Einleitung	32
2	Software	32
3	Hardware	33
3.1	Trackerschaltung	33
3.2	Geeignete GPS-Empfänger	33
4	Konfiguration	33
5	Referenzprojekte	34
5.1	OE5EEP, Heinz	34
6	Prämiert	35
7	LoRa Variante	36
8	Weitere Projekte	36

Einleitung

Der APRStracker von OE5DXL erlaubt es mit minimalstem Hardwareaufwand in der Betriebsart APRS QRV zu werden. Außerdem wurde bei der Entwicklung der Software darauf geachtet, die derzeitigen Möglichkeiten des APRS-Protokolls in Form von Mic-e optimalst auszunutzen. In der Praxis ist dies durch extrem kurze Frames erkennbar, was im Mobilbetrieb (QSB) erhebliche Vorteile bringt. Zu dem wird das schon etwas in die Jahre gekommene aber durchaus am effektivsten SSID-Pathrouting (im Configtool 'COMPRESSED' genannt) unterstützt.

Als Minimum an Hardware wird benötigt

- ein Mikrocontroller ATtiny13, ATtiny2313, Atmega88 oder auch andere Typen mit (Quarz)-Takt durch Anpassen der I/O Pins im Source
- Quarz frei wählbar ca. 6..20MHz (Im Source eingeben).
- PTT-Transistor
- RC-Tiefpass zum wegfiltern der PWM-Frequenz benötigt.

Software

Opensource Software von OE5DXL, in Assembler geschrieben, welche den Tracker zum Leben erweckt:

Datei:[AprsTracker.zip](#)

Der Assemblercode wird im einfachsten Fall mit dem Compiler 'gavrasm' [Gerd's AVR Assembler](#), welcher sowohl für Linux als auch für Windows verfügbar ist, kompiliert und anschließend in den µC gebrannt (z.B.: mit AVRdude [\[1\]](#) und dem USBasp Programmer [\[2\]](#)). Es ist auch möglich mit dem von ATMEL angebotenen AVR-Studio den Code zu übersetzen.

Fertige Kompilate:

Prozessor	Systemtakt	GPS-Baud	AFSK Baud	Download
ATMEL Tiny13	10 Mhz	4800	1200	Datei:Dxl-aprsTracker-tiny13-10MHz-4800Bd.zip
ATMEL Tiny13	10 Mhz	9600	1200	Datei:Dxl-aprsTracker-tiny13-10MHz-9600Bd.zip

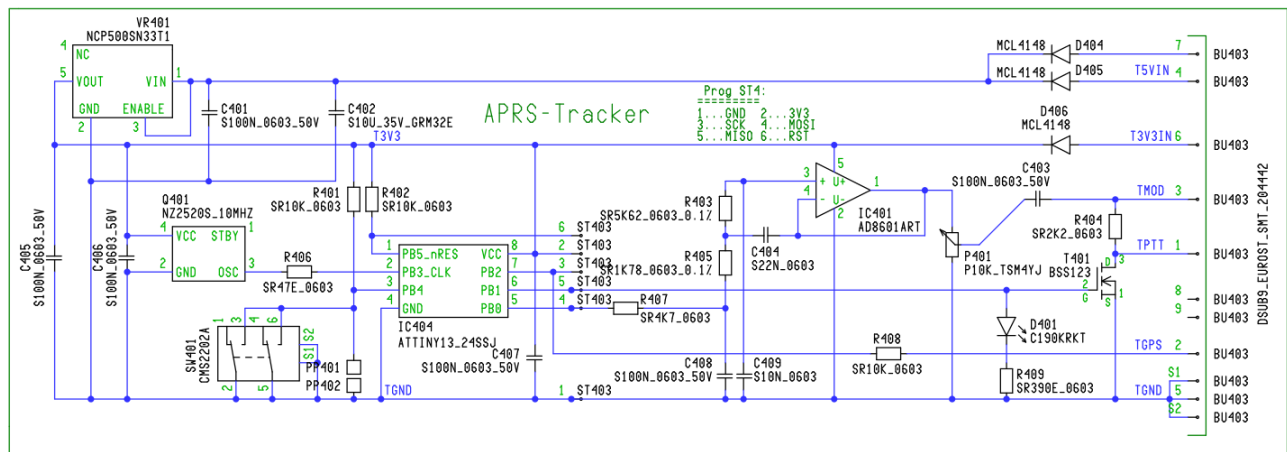
Als Input dienen serielle GPS-Daten (GPRMC und GPGLA), der Tracker generiert daraus anhand der programmierten Konfiguration als Output (nebst PTT) die AFSK-Modulation im APRS-Mic-e Format mit Position, Geschwindigkeit, Fahrtrichtung und Höhe, welche dem Funkgerät zugeführt wird.

Dabei sind (pro Profil) 2 Bakenzeiten und eine Geschwindigkeit einstellbar unter/ueber der langsam/schnell gebakt wird.

Hardware

Trackerschaltung

Schaltungsvorschlag von OE5HPM mit einem Tiny13 (kleinster Prozessor), wie er bereits mehrfach im Einsatz ist:



Detaillierte Schaltungsbeschreibung folgt.

Geeignete GPS-Empfänger

Hersteller	Type	Baudrate	Versorgungsspannung	Stromaufnahme	Preis	
Fastrax	UP501	9600	3.3V	~30mA	ca. 22€ (inkl. Versand)	http://clo.../Se...http://da.../UP...pdf
Ublox	NEO-6M	9600	3.3V	~22mA	ca. 10€ (inkl. Versand)	Ubl...
Globaltop	GTPA013	9600	3.3V	~20mA	ca. 19€	http://da...FGI...VO...

Konfiguration

GPS und Config-Programm liefern die seriellen Daten in TTL- oder RS232-Pegel. Der Pegel wird mittels einem (10k) Widerstand und der im Prozessor eingebauten Schutzdiode angepasst, die Polarisation erkennt die Software automatisch.

Baudraten vom GPS bzw. PC zur Konfiguration sind je nach Prozessortakt im Bereich 300..200000, ueblich 4800, 9600.

AFSK-Baud und Tonfrequenzen (Shift) sind ebenso frei Konfigurierbar, gebräuchlich sind:

- 300 auf Kurzwelle
- 1200 auf UKW

Zur Einstellung von HUB bzw. Mikrofonpegel eignet sich am besten ein Poti.

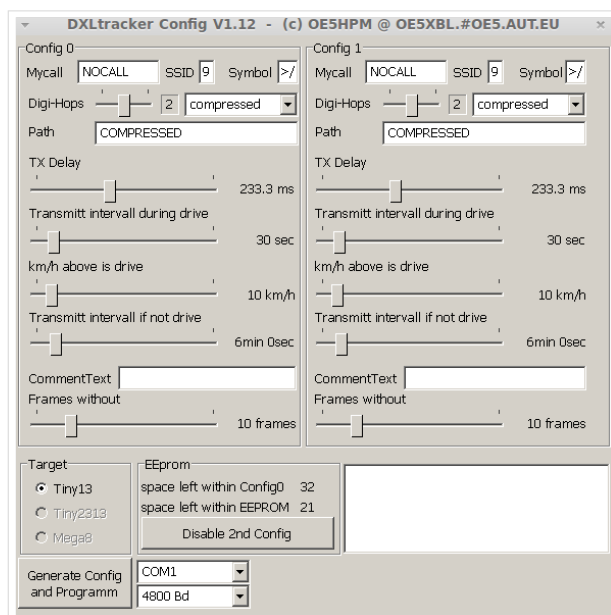
Um die PTT von Handfunkgeräten aufzutasten muss der Mikrofoneingang mit dem PTT Transistor kombiniert werden. Einfachste Variante ist im obigen Schaltungsbild sichtbar, der FET T402 zieht über einen 2k2 Widerstand den MIC-Eingang vom Handfunk gegen Masse und aktiviert dadurch die PTT, über C403 (100nF) wird die Modulation "eingekoppelt".

Als Option kann mit einem Jumper/Schalter (im Schaltbild SW401) zwischen 2 (am Tiny13) oder 4 (auf größeren Prozessoren) User-Profilen ausgewählt werden, zB. Fahrrad/Auto.

An einer Blink-Led für korrekten GPS-Empfang am Prozessor Pin wird gearbeitet.

Zur Konfiguration kommt ein kleines Tool von OE5HPM - DXLtrackerConfig zum Einsatz, dies generiert einen Konfigurationsstring für den DXLtracker und schickt selbigen über die serielle Schnittstelle raus.

Datei: [DXLtrackerConfig.zip](#)



Screenshot DXL-Trackerconfig

Referenzprojekte

OE5EEP, Heinz

Mein Aufbau eines DXL Modems mit einem Fastrax UP501 GPS Empfänger, den ich für Bergtouren im Rahmen des SOTA-Programms einsetzen möchte: Das DXL Modem wurde mir freundlicherweise vom ADL501 zur Verfügung gestellt, der UP501 ist über e-Bay zu beziehen. Als Spannungsquelle für beide Baugruppen gemeinsam dienen 4 NiMH AAA-Zellen von Conrad mit einer nominalen Kapazität von 1100mAh, was für einen ganzen Bergtag leicht ausreicht. Die stabilisierte Spannung von 3,3V für des GPS Modul kann nach Brückung einer Diode an Pin 7 des Sub-D Anschlusses des DXL Trackers abgegriffen werden.

Da ich der Verlässlichkeit von freien Verkabelungen nicht traue, hab ich das GPS Modul mittels der mitgelieferten Steckerleiste auf einer Lochrasterplatte aufgelötet und die Verkabelungen zum DXL Modem auf beiden Seiten der Lochrasterplatte auf eine 9-polige Sub-D Buchse geführt. Dazu kommt noch ein Aus/Ein Schalter und ein Halter für die NiMH Zellen. Eingebaut hab ich das Ganze in eine ehemalige Präsentationsverpackung von 2 Kugelschreibern. Dieses transparente Kunststoffgehäuse erlaubt es, das Aufleuchten der LED im Sendefall von außen zu erkennen und

passt von der Größe her gut zum verwendeten Handfunkgerät (siehe Fotos). Der Aufbau wird am Handfunkgerät einfach mit Gummibändern (Stücke von einem Fahrradschlauch) befestigt. Die einzige elektrische Verbindung zum Handfunkgerät ist eine Audioleitung zur Mikrofonbuchse, in meinem Fall mit einem 2,5mm Mono-Klinkenstecker.

Bisher hab ich den Tracker auf mehreren Bergtouren eingesetzt und positive Erfahrungen gemacht. Der GPS Empfänger findet binnen weniger Minuten eine Position. Die Empfindlichkeit ist ausgezeichnet. Teilweise kann ich im Inneren von Gebäuden eine Position bekommen, allerdings dann mit größerem Fehler. Jedenfalls ist der GPS-Empfänger deutlich empfindlicher als der in meinem LG Smartphone. Im Zusammenspiel mit einem schon etwas älteren IC-2e mit etwa 3W Sendeleistung und Gummiantenne ist es mir auf Touren in den Kalkalpen immer gelungen APRS-Meldungen im Gipfelbereich abzusetzen. Im Funkschatten von Bergen verlief das nicht so verlässlich. Das ist aber kein Problem der Kombination Tracker und Handfunkgerät sondern ist auf den dünnen Ausbau der sonstige APRS Infrastruktur zurückzuführen.

Dieses Selbstbauprojekt ist sowohl technisch als auch finanziell überschaubar. Ich hoffe, dass genaue Positionsmeldungen während meiner Bergtouren die SOTA Jäger unterstützen und gegebenenfalls zu meiner Sicherheit in den Bergen beitragen! Danke an OE5DXL für die Entwicklung der Basis für dieses nette Selbstbauprojekt!

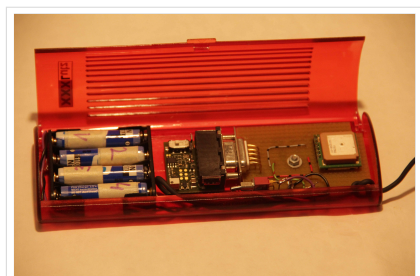
73 Heinz, OE5EEP



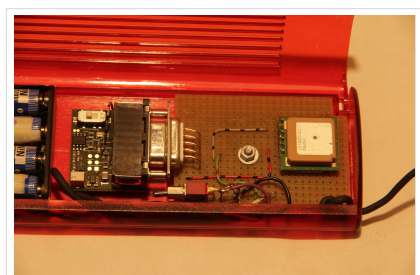
Gesamtansicht



Gesamtansicht



Innenleben



Innenleben-Detail-Tracker+GPS-RX

Prämiert

Der DXL APRStracker wurde mit dem € 2000,- dotierten ÖVSV Innovationspreis 2013 prämiert !
GRATULATION !!

LoRa Variante

Von OE5DXL gibt es auch eine LoRa Variante für Hardware-Tracker des Typs TTGO bzw. LILYGO inkl. Unterstützung für komprimierte Aussendung und Verwendung von MIC-E.

Zu finden ist die fertige INO-Datei in verschiedenen Ausführung, neben anderer Software, wie immer im [HAMNET Indexserver von OE5DXL](#).

Weitere Projekte

Weiter passende Projekte von OE5DXL aus dieser Serie sind

- [DXL - APRSmap](#) - Der neue APRS Client für Windows, Linux und ARM auf OSM Basis
- [TCE Tyncore Linux Projekt](#) - Der mächtige und innovative Digipeater für APRS, Packet Radio, Echolink, u.v.m.

Dateiversionen

Klicken Sie auf einen Zeitpunkt, um diese Version zu laden.

	Version vom	Maße	Benutzer	Kommentar
aktuell	21:13, 28. Aug. 2013	(7 KB)	OE5HPM (Diskussion Beiträge)	

Sie können diese Datei nicht überschreiben.

Dateiverwendung

Die folgende Seite verwendet diese Datei:

- [DXL - APRStracker](#)

DXL - APRTracker: Unterschied zwischen den Versionen

[Versionsgeschichte interaktiv durchsuchen](#)

[Visuell Wikitext](#)

Version vom 24. September 2023, 20:11

Uhr (Quelltext anzeigen)

[OE2WAO](#) ([Diskussion](#) | [Beiträge](#))

K

Markierung: 2017-Quelltext-Bearbeitung

[← Zum vorherigen Versionsunterschied](#)

Version vom 24. September 2023, 20:13

Uhr (Quelltext anzeigen)

[OE2WAO](#) ([Diskussion](#) | [Beiträge](#))

K

Markierung: 2017-Quelltext-Bearbeitung

[Zum nächsten Versionsunterschied →](#)

Zeile 32:

|4800

|1200

– [[Datei:**dxl**-aprsTracker-tiny13-10MHz-4800Bd.zip|verweis=Special:FilePath/dxl-aprsTracker-tiny13-10MHz-4800Bd.zip]]

|-

|ATMEL Tiny13

Zeile 32:

|4800

|1200

+ [[Datei:**Dxl**-aprsTracker-tiny13-10MHz-4800Bd.zip|verweis=Special:FilePath/dxl-aprsTracker-tiny13-10MHz-4800Bd.zip]]

|-

|ATMEL Tiny13

Zeile 38:

|9600

|1200

– [[Datei:**dxl**-aprsTracker-tiny13-10MHz-9600Bd.zip|verweis=Special:FilePath/dxl-aprsTracker-tiny13-10MHz-9600Bd.zip]]

|-

|}

Zeile 38:

|9600

|1200

+ [[Datei:**Dxl**-aprsTracker-tiny13-10MHz-9600Bd.zip|verweis=Special:FilePath/dxl-aprsTracker-tiny13-10MHz-9600Bd.zip]]

|-

|}

Zeile 122:

[[Datei:Oe5eep_2.jpg|rechts|mini|hochkant|200px|Gesamtansicht|verweis=Special:FilePath/oe5eep_2.jpg]]

– [[Datei:**oe5eep_1**.jpg|rechts|mini|200px|Gesamtansicht|verweis=Special:FilePath/oe5eep_1.jpg]]

Zeile 122:

[[Datei:Oe5eep_2.jpg|rechts|mini|hochkant|200px|Gesamtansicht|verweis=Special:FilePath/oe5eep_2.jpg]]

+ [[Datei:**Oe5eep_1**.jpg|rechts|mini|200px|Gesamtansicht|verweis=Special:FilePath/oe5eep_1.jpg]]

-	[[Datei:oe5eep_3.jpg rechts mini 200px Innenleben verweis=Special:FilePath/oe5eep_3.jpg]]	+	[[Datei:Oe5eep_3.jpg rechts mini 200px Innenleben verweis=Special:FilePath/oe5eep_3.jpg]]
-	[[Datei:oe5eep_4h.jpg rechts mini 200px Innenleben-Detail-Tracker+GPS-RX verweis=Special:FilePath/oe5eep_4h.jpg]]	+	[[Datei:Oe5eep_4h.jpg rechts mini 200px Innenleben-Detail-Tracker+GPS-RX verweis=Special:FilePath/oe5eep_4h.jpg]]

Version vom 24. September 2023, 20:13 Uhr

Inhaltsverzeichnis

1	Einleitung	39
2	Software	39
3	Hardware	40
3.1	Trackerschaltung	40
3.2	Geeignete GPS-Empfänger	40
4	Konfiguration	40
5	Referenzprojekte	41
5.1	OE5EEP, Heinz	41
6	Prämiert	42
7	LoRa Variante	43
8	Weitere Projekte	43

Einleitung

Der APRStracker von OE5DXL erlaubt es mit minimalstem Hardwareaufwand in der Betriebsart APRS QRV zu werden. Außerdem wurde bei der Entwicklung der Software darauf geachtet, die derzeitigen Möglichkeiten des APRS-Protokolls in Form von Mic-e optimalst auszunutzen. In der Praxis ist dies durch extrem kurze Frames erkennbar, was im Mobilbetrieb (QSB) erhebliche Vorteile bringt. Zu dem wird das schon etwas in die Jahre gekommene aber durchaus am effektivsten SSID-Pathrouting (im Configtool 'COMPRESSED' genannt) unterstützt.

Als Minimum an Hardware wird benötigt

- ein Mikrocontroller ATtiny13, ATtiny2313, Atmega88 oder auch andere Typen mit (Quarz)-Takt durch Anpassen der I/O Pins im Source
- Quarz frei wählbar ca. 6..20MHz (Im Source eingeben).
- PTT-Transistor
- RC-Tiefpass zum wegfiltern der PWM-Frequenz benötigt.

Software

Opensource Software von OE5DXL, in Assembler geschrieben, welche den Tracker zum Leben erweckt:

Datei:[AprsTracker.zip](#)

Der Assemblercode wird im einfachsten Fall mit dem Compiler 'gavrasm' [Gerd's AVR Assembler](#), welcher sowohl für Linux als auch für Windows verfügbar ist, kompiliert und anschließend in den µC gebrannt (z.B.: mit AVRdude [\[1\]](#) und dem USBasp Programmer [\[2\]](#)). Es ist auch möglich mit dem von ATMEL angebotenen AVR-Studio den Code zu übersetzen.

Fertige Kompilate:

Prozessor	Systemtakt	GPS-Baud	AFSK Baud	Download
ATMEL Tiny13	10 Mhz	4800	1200	Datei:Dxl-aprsTracker-tiny13-10MHz-4800Bd.zip
ATMEL Tiny13	10 Mhz	9600	1200	Datei:Dxl-aprsTracker-tiny13-10MHz-9600Bd.zip

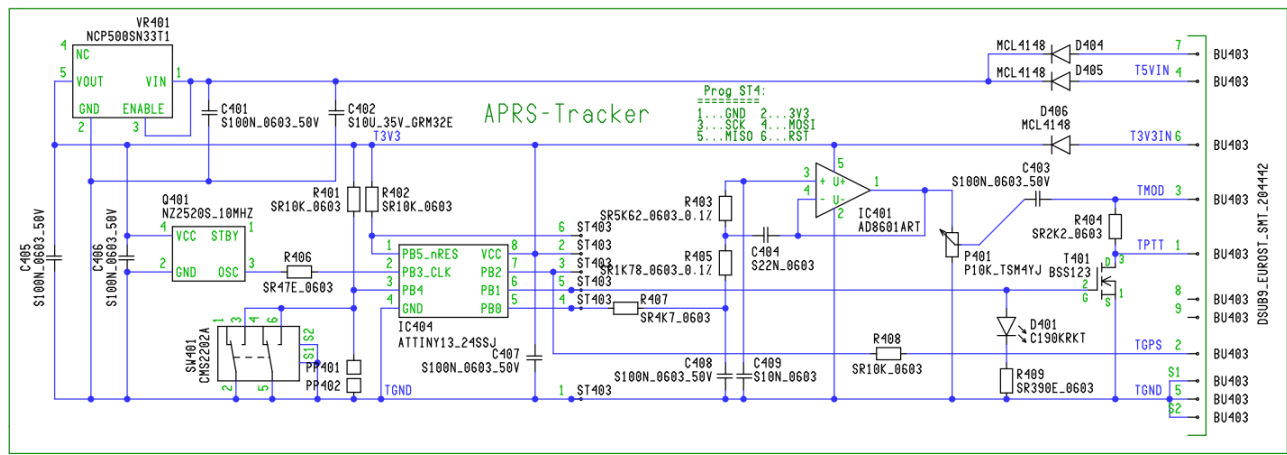
Als Input dienen serielle GPS-Daten (GPRMC und GPGLL), der Tracker generiert daraus anhand der programmierten Konfiguration als Output (nebst PTT) die AFSK-Modulation im APRS-Mic-e Format mit Position, Geschwindigkeit, Fahrtrichtung und Höhe, welche dem Funkgerät zugeführt wird.

Dabei sind (pro Profil) 2 Bakenzeiten und eine Geschwindigkeit einstellbar unter/ueber der langsam/schnell gebakt wird.

Hardware

Trackerschaltung

Schaltungsvorschlag von OE5HPM mit einem Tiny13 (kleinster Prozessor), wie er bereits mehrfach im Einsatz ist:



Detaillierte Schaltungsbeschreibung folgt.

Geeignete GPS-Empfänger

Hersteller	Type	Baudrate	Versorgungsspannung	Stromaufnahme	Preis	
Fastrax	UP501	9600	3.3V	~30mA	ca. 22€ (inkl. Versand)	http://clo.../Se...http://da.../UP...pdf
Ublox	NEO-6M	9600	3.3V	~22mA	ca. 10€ (inkl. Versand)	Ubl...
Globaltop	GTPA013	9600	3.3V	~20mA	ca. 19€	http://da...FGI...VO...

Konfiguration

GPS und Config-Programm liefern die seriellen Daten in TTL- oder RS232-Pegel. Der Pegel wird mittels einem (10k) Widerstand und der im Prozessor eingebauten Schutzdiode angepasst, die Polarisation erkennt die Software automatisch.

Baudraten vom GPS bzw. PC zur Konfiguration sind je nach Prozessortakt im Bereich 300..200000, ueblich 4800, 9600.

AFSK-Baud und Tonfrequenzen (Shift) sind ebenso frei Konfigurierbar, gebräuchlich sind:

- 300 auf Kurzwelle
- 1200 auf UKW

Zur Einstellung von HUB bzw. Mikrofonpegel eignet sich am besten ein Poti.

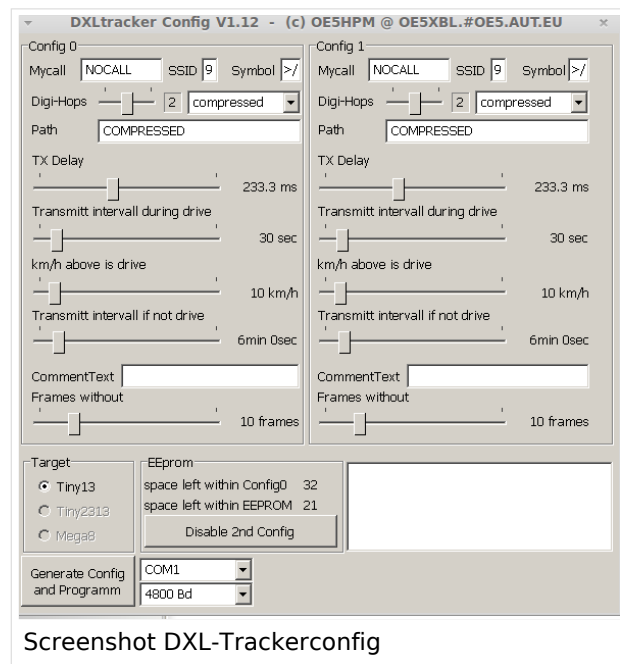
Um die PTT von Handfunkgeräten aufzutasten muss der Mikrofoneingang mit dem PTT Transistor kombiniert werden. Einfachste Variante ist im obigen Schaltungsbild sichtbar, der FET T402 zieht über einen 2k2 Widerstand den MIC-Eingang vom Handfunk gegen Masse und aktiviert dadurch die PTT, über C403 (100nF) wird die Modulation "eingekoppelt".

Als Option kann mit einem Jumper/Schalter (im Schaltbild SW401) zwischen 2 (am Tiny13) oder 4 (auf größeren Prozessoren) User-Profilen ausgewählt werden, zB. Fahrrad/Auto.

An einer Blink-Led für korrekten GPS-Empfang am Prozessor Pin wird gearbeitet.

Zur Konfiguration kommt ein kleines Tool von OE5HPM - DXLtrackerConfig zum Einsatz, dies generiert einen Konfigurationsstring für den DXLtracker und schickt selbigen über die serielle Schnittstelle raus.

Datei: [DXLtrackerConfig.zip](#)



Screenshot DXL-Trackerconfig

Referenzprojekte

OE5EEP, Heinz

Mein Aufbau eines DXL Modems mit einem Fastrax UP501 GPS Empfänger, den ich für Bergtouren im Rahmen des SOTA-Programms einsetzen möchte: Das DXL Modem wurde mir freundlicherweise vom ADL501 zur Verfügung gestellt, der UP501 ist über e-Bay zu beziehen. Als Spannungsquelle für beide Baugruppen gemeinsam dienen 4 NiMH AAA-Zellen von Conrad mit einer nominalen Kapazität von 1100mAh, was für einen ganzen Bergtag leicht ausreicht. Die stabilisierte Spannung von 3,3V für des GPS Modul kann nach Brückung einer Diode an Pin 7 des Sub-D Anschlusses des DXL Trackers abgegriffen werden.

Da ich der Verlässlichkeit von freien Verkabelungen nicht traue, hab ich das GPS Modul mittels der mitgelieferten Steckerleiste auf einer Lochrasterplatte aufgelötet und die Verkabelungen zum DXL Modem auf beiden Seiten der Lochrasterplatte auf eine 9-polige Sub-D Buchse geführt. Dazu kommt noch ein Aus/Ein Schalter und ein Halter für die NiMH Zellen. Eingebaut hab ich das Ganze in eine ehemalige Präsentationsverpackung von 2 Kugelschreibern. Dieses transparente Kunststoffgehäuse erlaubt es, das Aufleuchten der LED im Sendefall von außen zu erkennen und

passt von der Größe her gut zum verwendeten Handfunkgerät (siehe Fotos). Der Aufbau wird am Handfunkgerät einfach mit Gummibändern (Stücke von einem Fahrradschlauch) befestigt. Die einzige elektrische Verbindung zum Handfunkgerät ist eine Audioleitung zur Mikrofonbuchse, in meinem Fall mit einem 2,5mm Mono-Klinkenstecker.

Bisher hab ich den Tracker auf mehreren Bergtouren eingesetzt und positive Erfahrungen gemacht. Der GPS Empfänger findet binnen weniger Minuten eine Position. Die Empfindlichkeit ist ausgezeichnet. Teilweise kann ich im Inneren von Gebäuden eine Position bekommen, allerdings dann mit größerem Fehler. Jedenfalls ist der GPS-Empfänger deutlich empfindlicher als der in meinem LG Smartphone. Im Zusammenspiel mit einem schon etwas älteren IC-2e mit etwa 3W Sendeleistung und Gummiantenne ist es mir auf Touren in den Kalkalpen immer gelungen APRS-Meldungen im Gipfelbereich abzusetzen. Im Funkschatten von Bergen verlief das nicht so verlässlich. Das ist aber kein Problem der Kombination Tracker und Handfunkgerät sondern ist auf den dünnen Ausbau der sonstige APRS Infrastruktur zurückzuführen.

Dieses Selbstbauprojekt ist sowohl technisch als auch finanziell überschaubar. Ich hoffe, dass genaue Positionsmeldungen während meiner Bergtouren die SOTA Jäger unterstützen und gegebenenfalls zu meiner Sicherheit in den Bergen beitragen! Danke an OE5DXL für die Entwicklung der Basis für dieses nette Selbstbauprojekt!

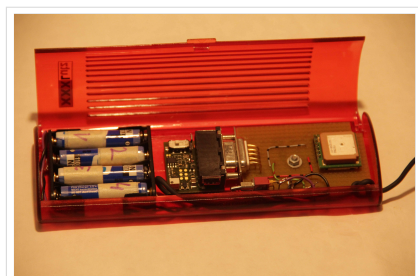
73 Heinz, OE5EEP



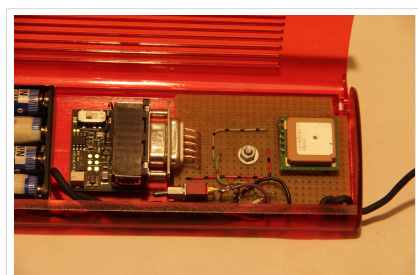
Gesamtansicht



Gesamtansicht



Innenleben



Innenleben-Detail-Tracker+GPS-RX

Prämiert

Der DXL APRStracker wurde mit dem € 2000,- dotierten ÖVSV Innovationspreis 2013 prämiert !
GRATULATION !!

LoRa Variante

Von OE5DXL gibt es auch eine LoRa Variante für Hardware-Tracker des Typs TTGO bzw. LILYGO inkl. Unterstützung für komprimierte Aussendung und Verwendung von MIC-E.

Zu finden ist die fertige INO-Datei in verschiedenen Ausführung, neben anderer Software, wie immer im [HAMNET Indexserver von OE5DXL](#).

Weitere Projekte

Weiter passende Projekte von OE5DXL aus dieser Serie sind

- [DXL - APRSmap](#) - Der neue APRS Client für Windows, Linux und ARM auf OSM Basis
- [TCE Tyncore Linux Projekt](#) - Der mächtige und innovative Digipeater für APRS, Packet Radio, Echolink, u.v.m.

Dateiversionen

Klicken Sie auf einen Zeitpunkt, um diese Version zu laden.

	Version vom	Maße	Benutzer	Kommentar
aktuell	20:44, 29. Sep. 2013	(48 KB)	OE5HPM (Diskussion) (Beiträge)	(V1.15.- SSID with more than 2 digits didn't work properly)
	21:21, 3. Sep. 2013	(48 KB)	OE5HPM (Diskussion) (Beiträge)	V1.14 - calls with len < 6 digits didn't work properly
	21:00, 28. Aug. 2013	(48 KB)	OE5HPM (Diskussion) (Beiträge)	V1.12 - letzte Releaseversion

Sie können diese Datei nicht überschreiben.

Dateiverwendung

Die folgende Seite verwendet diese Datei:

- [DXL - APRStracker](#)

DXL - APRTracker: Unterschied zwischen den Versionen

Versionsgeschichte interaktiv durchsuchen

VisuellWikitext

Version vom 24. September 2023, 20:11

Uhr (Quelltext anzeigen)

OE2WAO ([Diskussion](#) | [Beiträge](#))

K

Markierung: 2017-Quelltext-Bearbeitung

← Zum vorherigen Versionsunterschied

Version vom 24. September 2023, 20:13

Uhr (Quelltext anzeigen)

OE2WAO ([Diskussion](#) | [Beiträge](#))

K

Markierung: 2017-Quelltext-Bearbeitung

Zum nächsten Versionsunterschied →

Zeile 32:

|4800

|1200

– [[Datei:**dxl**-aprsTracker-tiny13-10MHz-4800Bd.zip|verweis=Special:FilePath/dxl-aprsTracker-tiny13-10MHz-4800Bd.zip]]

|-

|ATMEL Tiny13

Zeile 32:

|4800

|1200

+ [[Datei:**Dxl**-aprsTracker-tiny13-10MHz-4800Bd.zip|verweis=Special:FilePath/dxl-aprsTracker-tiny13-10MHz-4800Bd.zip]]

|-

|ATMEL Tiny13

Zeile 38:

|9600

|1200

– [[Datei:**dxl**-aprsTracker-tiny13-10MHz-9600Bd.zip|verweis=Special:FilePath/dxl-aprsTracker-tiny13-10MHz-9600Bd.zip]]

|-

|}

Zeile 38:

|9600

|1200

+ [[Datei:**Dxl**-aprsTracker-tiny13-10MHz-9600Bd.zip|verweis=Special:FilePath/dxl-aprsTracker-tiny13-10MHz-9600Bd.zip]]

|-

|}

Zeile 122:

[[Datei:Oe5eep_2.jpg|rechts|mini|hochkant|200px|Gesamtansicht|verweis=Special:FilePath/oe5eep_2.jpg]]

– [[Datei:**oe5eep_1**.jpg|rechts|mini|200px|Gesamtansicht|verweis=Special:FilePath/oe5eep_1.jpg]]

Zeile 122:

[[Datei:Oe5eep_2.jpg|rechts|mini|hochkant|200px|Gesamtansicht|verweis=Special:FilePath/oe5eep_2.jpg]]

+ [[Datei:**Oe5eep_1**.jpg|rechts|mini|200px|Gesamtansicht|verweis=Special:FilePath/oe5eep_1.jpg]]

-	[[Datei:oe5eep_3.jpg rechts mini 200px Innenleben verweis=Special:FilePath/oe5eep_3.jpg]]	+	[[Datei:Oe5eep_3.jpg rechts mini 200px Innenleben verweis=Special:FilePath/oe5eep_3.jpg]]
-	[[Datei:oe5eep_4h.jpg rechts mini 200px Innenleben-Detail-Tracker+GPS-RX verweis=Special:FilePath/oe5eep_4h.jpg]]	+	[[Datei:Oe5eep_4h.jpg rechts mini 200px Innenleben-Detail-Tracker+GPS-RX verweis=Special:FilePath/oe5eep_4h.jpg]]

Version vom 24. September 2023, 20:13 Uhr

Inhaltsverzeichnis

1	Einleitung	46
2	Software	46
3	Hardware	47
3.1	Trackerschaltung	47
3.2	Geeignete GPS-Empfänger	47
4	Konfiguration	47
5	Referenzprojekte	48
5.1	OE5EEP, Heinz	48
6	Prämiert	49
7	LoRa Variante	50
8	Weitere Projekte	50

Einleitung

Der APRStracker von OE5DXL erlaubt es mit minimalstem Hardwareaufwand in der Betriebsart APRS QRV zu werden. Außerdem wurde bei der Entwicklung der Software darauf geachtet, die derzeitigen Möglichkeiten des APRS-Protokolls in Form von Mic-e optimalst auszunutzen. In der Praxis ist dies durch extrem kurze Frames erkennbar, was im Mobilbetrieb (QSB) erhebliche Vorteile bringt. Zu dem wird das schon etwas in die Jahre gekommene aber durchaus am effektivsten SSID-Pathrouting (im Configtool 'COMPRESSED' genannt) unterstützt.

Als Minimum an Hardware wird benötigt

- ein Mikrocontroller ATtiny13, ATtiny2313, Atmega88 oder auch andere Typen mit (Quarz)-Takt durch Anpassen der I/O Pins im Source
- Quarz frei wählbar ca. 6..20MHz (Im Source eingeben).
- PTT-Transistor
- RC-Tiefpass zum wegfiltern der PWM-Frequenz benötigt.

Software

Opensource Software von OE5DXL, in Assembler geschrieben, welche den Tracker zum Leben erweckt:

Datei:[AprsTracker.zip](#)

Der Assemblercode wird im einfachsten Fall mit dem Compiler 'gavrasm' [Gerd's AVR Assembler](#), welcher sowohl für Linux als auch für Windows verfügbar ist, kompiliert und anschließend in den µC gebrannt (z.B.: mit AVRdude [\[1\]](#) und dem USBasp Programmer [\[2\]](#)). Es ist auch möglich mit dem von ATMEL angebotenen AVR-Studio den Code zu übersetzen.

Fertige Kompilate:

Prozessor	Systemtakt	GPS-Baud	AFSK Baud	Download
ATMEL Tiny13	10 Mhz	4800	1200	Datei:Dxl-aprsTracker-tiny13-10MHz-4800Bd.zip
ATMEL Tiny13	10 Mhz	9600	1200	Datei:Dxl-aprsTracker-tiny13-10MHz-9600Bd.zip

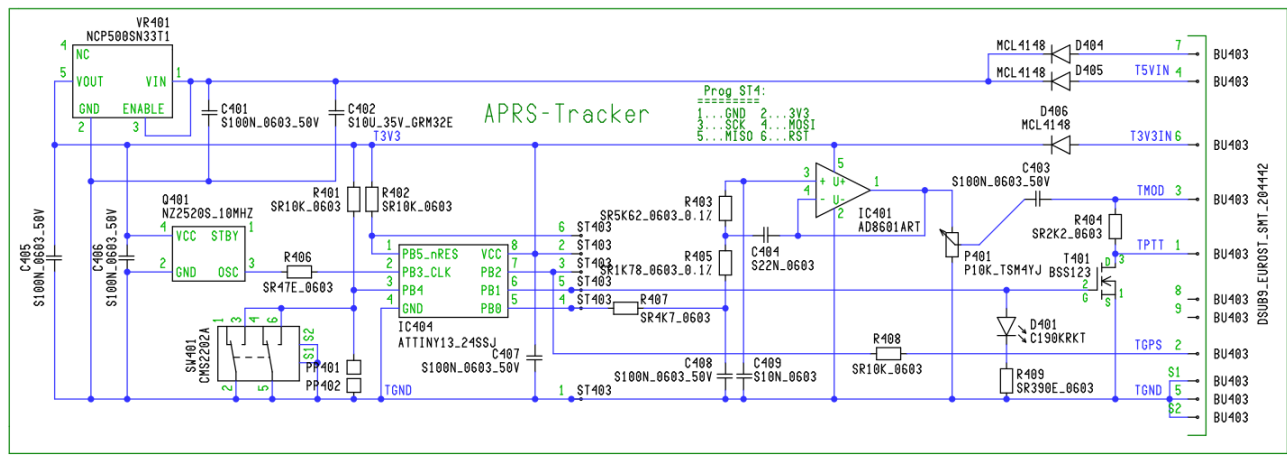
Als Input dienen serielle GPS-Daten (GPRMC und GPGLL), der Tracker generiert daraus anhand der programmierten Konfiguration als Output (nebst PTT) die AFSK-Modulation im APRS-Mic-e Format mit Position, Geschwindigkeit, Fahrtrichtung und Höhe, welche dem Funkgerät zugeführt wird.

Dabei sind (pro Profil) 2 Bakenzeiten und eine Geschwindigkeit einstellbar unter/ueber der langsam/schnell gebakt wird.

Hardware

Trackerschaltung

Schaltungsvorschlag von OE5HPM mit einem Tiny13 (kleinster Prozessor), wie er bereits mehrfach im Einsatz ist:



Detaillierte Schaltungsbeschreibung folgt.

Geeignete GPS-Empfänger

Hersteller	Type	Baudrate	Versorgungsspannung	Stromaufnahme	Preis	
Fastrax	UP501	9600	3.3V	~30mA	ca. 22€ (inkl. Versand)	http://clo.../Se...http://da.../UP.pdf
Ublox	NEO-6M	9600	3.3V	~22mA	ca. 10€ (inkl. Versand)	Ubl...
Globaltop	GTPA013	9600	3.3V	~20mA	ca. 19€	http://da...FGI...VO/

Konfiguration

GPS und Config-Programm liefern die seriellen Daten in TTL- oder RS232-Pegel. Der Pegel wird mittels einem (10k) Widerstand und der im Prozessor eingebauten Schutzdiode angepasst, die Polarisation erkennt die Software automatisch.

Baudraten vom GPS bzw. PC zur Konfiguration sind je nach Prozessortakt im Bereich 300..200000, ueblich 4800, 9600.

AFSK-Baud und Tonfrequenzen (Shift) sind ebenso frei Konfigurierbar, gebräuchlich sind:

- 300 auf Kurzwelle
- 1200 auf UKW

Zur Einstellung von HUB bzw. Mikrofonpegel eignet sich am besten ein Poti.

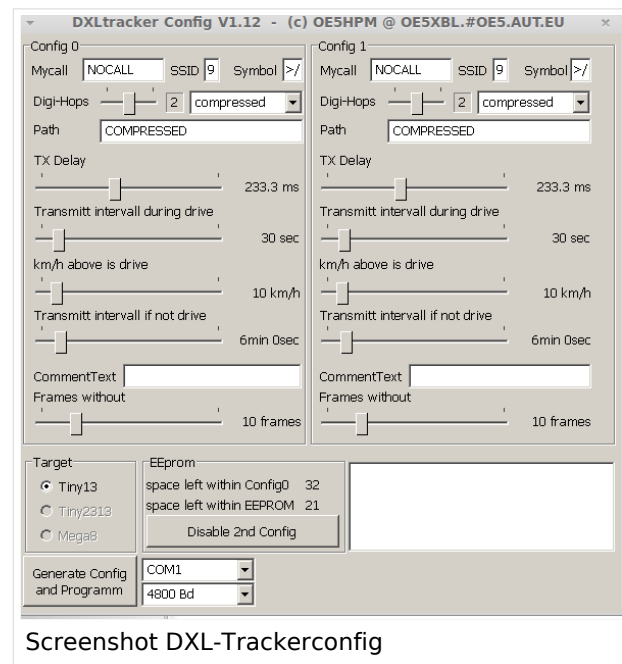
Um die PTT von Handfunkgeräten aufzutasten muss der Mikrofoneingang mit dem PTT Transistor kombiniert werden. Einfachste Variante ist im obigen Schaltungsbild sichtbar, der FET T402 zieht über einen 2k2 Widerstand den MIC-Eingang vom Handfunk gegen Masse und aktiviert dadurch die PTT, über C403 (100nF) wird die Modulation "eingekoppelt".

Als Option kann mit einem Jumper/Schalter (im Schaltbild SW401) zwischen 2 (am Tiny13) oder 4 (auf größeren Prozessoren) User-Profilen ausgewählt werden, zB. Fahrrad/Auto.

An einer Blink-Led für korrekten GPS-Empfang am Prozessor Pin wird gearbeitet.

Zur Konfiguration kommt ein kleines Tool von OE5HPM - DXLtrackerConfig zum Einsatz, dies generiert einen Konfigurationsstring für den DXLtracker und schickt selbigen über die serielle Schnittstelle raus.

[Datei:DXLtrackerConfig.zip](#)



Screenshot DXL-Trackerconfig

Referenzprojekte

OE5EEP, Heinz

Mein Aufbau eines DXL Modems mit einem Fastrax UP501 GPS Empfänger, den ich für Bergtouren im Rahmen des SOTA-Programms einsetzen möchte: Das DXL Modem wurde mir freundlicherweise vom ADL501 zur Verfügung gestellt, der UP501 ist über e-Bay zu beziehen. Als Spannungsquelle für beide Baugruppen gemeinsam dienen 4 NiMH AAA-Zellen von Conrad mit einer nominalen Kapazität von 1100mAh, was für einen ganzen Bergtag leicht ausreicht. Die stabilisierte Spannung von 3,3V für des GPS Modul kann nach Brückung einer Diode an Pin 7 des Sub-D Anschlusses des DXL Trackers abgegriffen werden.

Da ich der Verlässlichkeit von freien Verkabelungen nicht traue, hab ich das GPS Modul mittels der mitgelieferten Steckerleiste auf einer Lochrasterplatte aufgelötet und die Verkabelungen zum DXL Modem auf beiden Seiten der Lochrasterplatte auf eine 9-polige Sub-D Buchse geführt. Dazu kommt noch ein Aus/Ein Schalter und ein Halter für die NiMH Zellen. Eingebaut hab ich das Ganze in eine ehemalige Präsentationsverpackung von 2 Kugelschreibern. Dieses transparente Kunststoffgehäuse erlaubt es, das Aufleuchten der LED im Sendefall von außen zu erkennen und

passt von der Größe her gut zum verwendeten Handfunkgerät (siehe Fotos). Der Aufbau wird am Handfunkgerät einfach mit Gummibändern (Stücke von einem Fahrradschlauch) befestigt. Die einzige elektrische Verbindung zum Handfunkgerät ist eine Audioleitung zur Mikrofonbuchse, in meinem Fall mit einem 2,5mm Mono-Klinkenstecker.

Bisher hab ich den Tracker auf mehreren Bergtouren eingesetzt und positive Erfahrungen gemacht. Der GPS Empfänger findet binnen weniger Minuten eine Position. Die Empfindlichkeit ist ausgezeichnet. Teilweise kann ich im Inneren von Gebäuden eine Position bekommen, allerdings dann mit größerem Fehler. Jedenfalls ist der GPS-Empfänger deutlich empfindlicher als der in meinem LG Smartphone. Im Zusammenspiel mit einem schon etwas älteren IC-2e mit etwa 3W Sendeleistung und Gummiantenne ist es mir auf Touren in den Kalkalpen immer gelungen APRS-Meldungen im Gipfelbereich abzusetzen. Im Funkschatten von Bergen verlief das nicht so verlässlich. Das ist aber kein Problem der Kombination Tracker und Handfunkgerät sondern ist auf den dünnen Ausbau der sonstige APRS Infrastruktur zurückzuführen.

Dieses Selbstbauprojekt ist sowohl technisch als auch finanziell überschaubar. Ich hoffe, dass genaue Positionsmeldungen während meiner Bergtouren die SOTA Jäger unterstützen und gegebenenfalls zu meiner Sicherheit in den Bergen beitragen! Danke an OE5DXL für die Entwicklung der Basis für dieses nette Selbstbauprojekt!

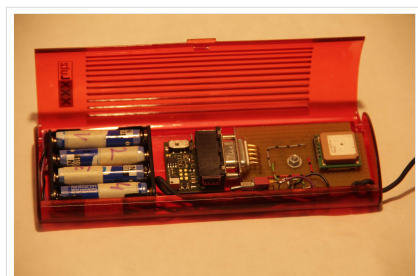
73 Heinz, OE5EEP



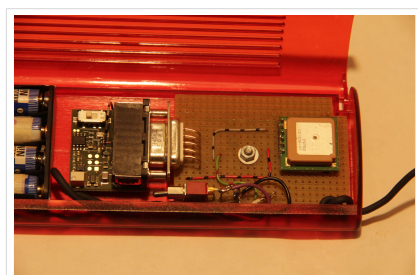
Gesamtansicht



Gesamtansicht



Innenleben



Innenleben-Detail-Tracker+GPS-RX

Prämiert

Der DXL APRStracker wurde mit dem € 2000,- dotierten ÖVSV Innovationspreis 2013 prämiert !
GRATULATION !!

LoRa Variante

Von OE5DXL gibt es auch eine LoRa Variante für Hardware-Tracker des Typs TTGO bzw. LILYGO inkl. Unterstützung für komprimierte Aussendung und Verwendung von MIC-E.

Zu finden ist die fertige INO-Datei in verschiedenen Ausführung, neben anderer Software, wie immer im [HAMNET Indexserver von OE5DXL](#).

Weitere Projekte

Weiter passende Projekte von OE5DXL aus dieser Serie sind

- [DXL - APRSmap](#) - Der neue APRS Client für Windows, Linux und ARM auf OSM Basis
- [TCE Tyncore Linux Projekt](#) - Der mächtige und innovative Digipeater für APRS, Packet Radio, Echolink, u.v.m.

Dateiversionen

Klicken Sie auf einen Zeitpunkt, um diese Version zu laden.

	Version vom	Maße	Benutzer	Kommentar
aktuell	21:46, 14. Sep. 2013	(9 KB)	OE5HPM (Diskussion)	Kompilat vom DXL aprsTracker für Tiny13 mit 10 MHz Systemtakt und 4800 Baud vom GPS-Empfänger, TX ist 1200Bd AFSK.

Sie können diese Datei nicht überschreiben.

Dateiverwendung

Die folgende Seite verwendet diese Datei:

- [DXL - APRStracker](#)

DXL - APRStracker: Unterschied zwischen den Versionen

Versionsgeschichte interaktiv durchsuchen

VisuellWikitext

Version vom 24. September 2023, 20:11

Uhr (Quelltext anzeigen)

OE2WAO ([Diskussion](#) | [Beiträge](#))

K

Markierung: 2017-Quelltext-Bearbeitung

← Zum vorherigen Versionsunterschied

Version vom 24. September 2023, 20:13

Uhr (Quelltext anzeigen)

OE2WAO ([Diskussion](#) | [Beiträge](#))

K

Markierung: 2017-Quelltext-Bearbeitung

Zum nächsten Versionsunterschied →

Zeile 32:

|4800

|1200

– [[Datei:**dxl**-aprsTracker-tiny13-10MHz-4800Bd.zip|verweis=Special:FilePath/dxl-aprsTracker-tiny13-10MHz-4800Bd.zip]]

|-

|ATMEL Tiny13

Zeile 32:

|4800

|1200

+ [[Datei:**Dxl**-aprsTracker-tiny13-10MHz-4800Bd.zip|verweis=Special:FilePath/dxl-aprsTracker-tiny13-10MHz-4800Bd.zip]]

|-

|ATMEL Tiny13

Zeile 38:

|9600

|1200

– [[Datei:**dxl**-aprsTracker-tiny13-10MHz-9600Bd.zip|verweis=Special:FilePath/dxl-aprsTracker-tiny13-10MHz-9600Bd.zip]]

|-

|}

Zeile 38:

|9600

|1200

+ [[Datei:**Dxl**-aprsTracker-tiny13-10MHz-9600Bd.zip|verweis=Special:FilePath/dxl-aprsTracker-tiny13-10MHz-9600Bd.zip]]

|-

|}

Zeile 122:

[[Datei:Oe5eep_2.jpg|rechts|mini|hochkant|200px|Gesamtansicht|verweis=Special:FilePath/oe5eep_2.jpg]]

– [[Datei:**oe5eep_1**.jpg|rechts|mini|200px|Gesamtansicht|verweis=Special:FilePath/oe5eep_1.jpg]]

Zeile 122:

[[Datei:Oe5eep_2.jpg|rechts|mini|hochkant|200px|Gesamtansicht|verweis=Special:FilePath/oe5eep_2.jpg]]

+ [[Datei:**Oe5eep_1**.jpg|rechts|mini|200px|Gesamtansicht|verweis=Special:FilePath/oe5eep_1.jpg]]

-	[[Datei:oe5eep_3.jpg rechts mini 200px Innenleben verweis=Special:FilePath/oe5eep_3.jpg]]	+	[[Datei:Oe5eep_3.jpg rechts mini 200px Innenleben verweis=Special:FilePath/oe5eep_3.jpg]]
-	[[Datei:oe5eep_4h.jpg rechts mini 200px Innenleben-Detail-Tracker+GPS-RX verweis=Special:FilePath/oe5eep_4h.jpg]]	+	[[Datei:Oe5eep_4h.jpg rechts mini 200px Innenleben-Detail-Tracker+GPS-RX verweis=Special:FilePath/oe5eep_4h.jpg]]

Version vom 24. September 2023, 20:13 Uhr

Inhaltsverzeichnis

1	Einleitung	53
2	Software	53
3	Hardware	54
3.1	Trackerschaltung	54
3.2	Geeignete GPS-Empfänger	54
4	Konfiguration	54
5	Referenzprojekte	55
5.1	OE5EEP, Heinz	55
6	Prämiert	56
7	LoRa Variante	57
8	Weitere Projekte	57

Einleitung

Der APRStracker von OE5DXL erlaubt es mit minimalstem Hardwareaufwand in der Betriebsart APRS QRV zu werden. Außerdem wurde bei der Entwicklung der Software darauf geachtet, die derzeitigen Möglichkeiten des APRS-Protokolls in Form von Mic-e optimalst auszunutzen. In der Praxis ist dies durch extrem kurze Frames erkennbar, was im Mobilbetrieb (QSB) erhebliche Vorteile bringt. Zu dem wird das schon etwas in die Jahre gekommene aber durchaus am effektivsten SSID-Pathrouting (im Configtool 'COMPRESSED' genannt) unterstützt.

Als Minimum an Hardware wird benötigt

- ein Mikrocontroller ATtiny13, ATtiny2313, Atmega88 oder auch andere Typen mit (Quarz)-Takt durch Anpassen der I/O Pins im Source
- Quarz frei wählbar ca. 6..20MHz (Im Source eingeben).
- PTT-Transistor
- RC-Tiefpass zum wegfiltern der PWM-Frequenz benötigt.

Software

Opensource Software von OE5DXL, in Assembler geschrieben, welche den Tracker zum Leben erweckt:

Datei:[AprsTracker.zip](#)

Der Assemblercode wird im einfachsten Fall mit dem Compiler 'gavrasm' [Gerd's AVR Assembler](#), welcher sowohl für Linux als auch für Windows verfügbar ist, kompiliert und anschließend in den µC gebrannt (z.B.: mit AVRdude [\[1\]](#) und dem USBasp Programmer [\[2\]](#)). Es ist auch möglich mit dem von ATMEL angebotenen AVR-Studio den Code zu übersetzen.

Fertige Kompilate:

Prozessor	Systemtakt	GPS-Baud	AFSK Baud	Download
ATMEL Tiny13	10 Mhz	4800	1200	Datei:Dxl-aprsTracker-tiny13-10MHz-4800Bd.zip
ATMEL Tiny13	10 Mhz	9600	1200	Datei:Dxl-aprsTracker-tiny13-10MHz-9600Bd.zip

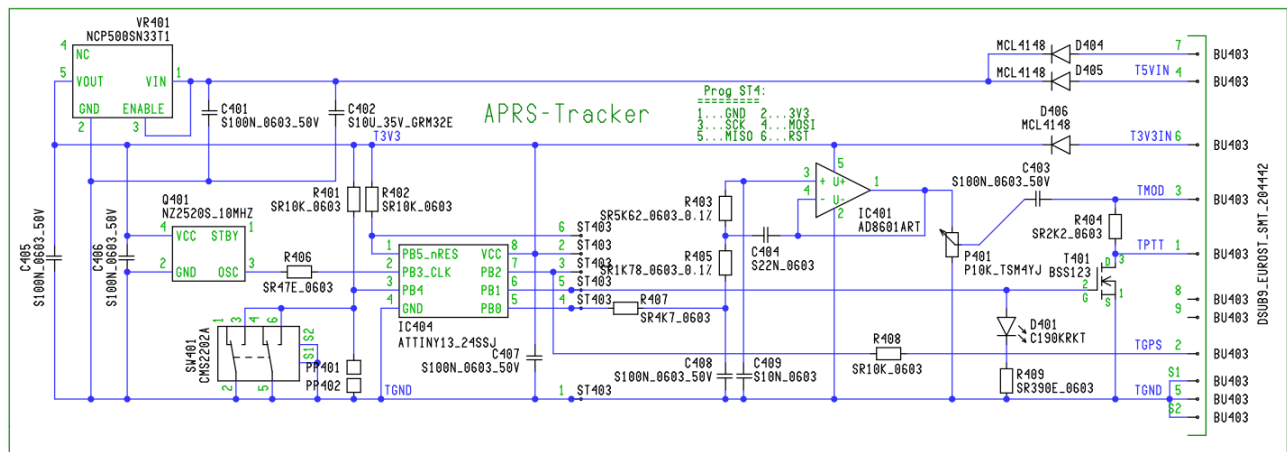
Als Input dienen serielle GPS-Daten (GPRMC und GPGLL), der Tracker generiert daraus anhand der programmierten Konfiguration als Output (nebst PTT) die AFSK-Modulation im APRS-Mic-e Format mit Position, Geschwindigkeit, Fahrtrichtung und Höhe, welche dem Funkgerät zugeführt wird.

Dabei sind (pro Profil) 2 Bakenzeiten und eine Geschwindigkeit einstellbar unter/ueber der langsam/schnell gebakt wird.

Hardware

Trackerschaltung

Schaltungsvorschlag von OE5HPM mit einem Tiny13 (kleinster Prozessor), wie er bereits mehrfach im Einsatz ist:



Detaillierte Schaltungsbeschreibung folgt.

Geeignete GPS-Empfänger

Hersteller	Type	Baudrate	Versorgungsspannung	Stromaufnahme	Preis	
Fastrax	UP501	9600	3.3V	~30mA	ca. 22€ (inkl. Versand)	http://clo.../Se...http://da.../UP...pdf
Ublox	NEO-6M	9600	3.3V	~22mA	ca. 10€ (inkl. Versand)	Ubl...
Globaltop	GTPA013	9600	3.3V	~20mA	ca. 19€	http://da...FGI...VO...

Konfiguration

GPS und Config-Programm liefern die seriellen Daten in TTL- oder RS232-Pegel. Der Pegel wird mittels einem (10k) Widerstand und der im Prozessor eingebauten Schutzdiode angepasst, die Polarisation erkennt die Software automatisch.

Baudraten vom GPS bzw. PC zur Konfiguration sind je nach Prozessortakt im Bereich 300..200000, ueblich 4800, 9600.

AFSK-Baud und Tonfrequenzen (Shift) sind ebenso frei Konfigurierbar, gebräuchlich sind:

- 300 auf Kurzwelle
- 1200 auf UKW

Zur Einstellung von HUB bzw. Mikrofonpegel eignet sich am besten ein Poti.

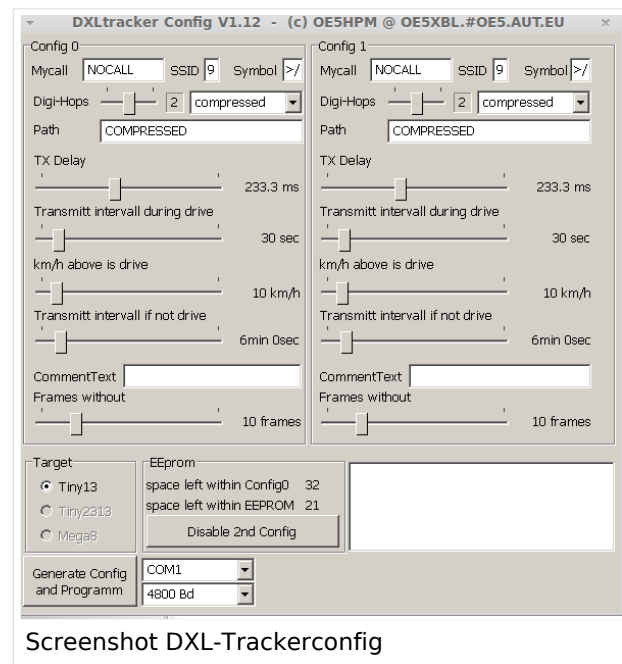
Um die PTT von Handfunkgeräten aufzutasten muss der Mikrofoneingang mit dem PTT Transistor kombiniert werden. Einfachste Variante ist im obigen Schaltungsbild sichtbar, der FET T402 zieht über einen 2k2 Widerstand den MIC-Eingang vom Handfunk gegen Masse und aktiviert dadurch die PTT, über C403 (100nF) wird die Modulation "eingekoppelt".

Als Option kann mit einem Jumper/Schalter (im Schaltbild SW401) zwischen 2 (am Tiny13) oder 4 (auf größeren Prozessoren) User-Profilen ausgewählt werden, zB. Fahrrad/Auto.

An einer Blink-Led für korrekten GPS-Empfang am Prozessor Pin wird gearbeitet.

Zur Konfiguration kommt ein kleines Tool von OE5HPM - DXLtrackerConfig zum Einsatz, dies generiert einen Konfigurationsstring für den DXLtracker und schickt selbigen über die serielle Schnittstelle raus.

Datei: [DXLtrackerConfig.zip](#)



Screenshot DXL-Trackerconfig

Referenzprojekte

OE5EEP, Heinz

Mein Aufbau eines DXL Modems mit einem Fastrax UP501 GPS Empfänger, den ich für Bergtouren im Rahmen des SOTA-Programms einsetzen möchte: Das DXL Modem wurde mir freundlicherweise vom ADL501 zur Verfügung gestellt, der UP501 ist über e-Bay zu beziehen. Als Spannungsquelle für beide Baugruppen gemeinsam dienen 4 NiMH AAA-Zellen von Conrad mit einer nominalen Kapazität von 1100mAh, was für einen ganzen Bergtag leicht ausreicht. Die stabilisierte Spannung von 3,3V für des GPS Modul kann nach Brückung einer Diode an Pin 7 des Sub-D Anschlusses des DXL Trackers abgegriffen werden.

Da ich der Verlässlichkeit von freien Verkabelungen nicht traue, hab ich das GPS Modul mittels der mitgelieferten Steckerleiste auf einer Lochrasterplatte aufgelötet und die Verkabelungen zum DXL Modem auf beiden Seiten der Lochrasterplatte auf eine 9-polige Sub-D Buchse geführt. Dazu kommt noch ein Aus/Ein Schalter und ein Halter für die NiMH Zellen. Eingebaut hab ich das Ganze in eine ehemalige Präsentationsverpackung von 2 Kugelschreibern. Dieses transparente Kunststoffgehäuse erlaubt es, das Aufleuchten der LED im Sendefall von außen zu erkennen und

passt von der Größe her gut zum verwendeten Handfunkgerät (siehe Fotos). Der Aufbau wird am Handfunkgerät einfach mit Gummibändern (Stücke von einem Fahrradschlauch) befestigt. Die einzige elektrische Verbindung zum Handfunkgerät ist eine Audioleitung zur Mikrofonbuchse, in meinem Fall mit einem 2,5mm Mono-Klinkenstecker.

Bisher hab ich den Tracker auf mehreren Bergtouren eingesetzt und positive Erfahrungen gemacht. Der GPS Empfänger findet binnen weniger Minuten eine Position. Die Empfindlichkeit ist ausgezeichnet. Teilweise kann ich im Inneren von Gebäuden eine Position bekommen, allerdings dann mit größerem Fehler. Jedenfalls ist der GPS-Empfänger deutlich empfindlicher als der in meinem LG Smartphone. Im Zusammenspiel mit einem schon etwas älteren IC-2e mit etwa 3W Sendeleistung und Gummiantenne ist es mir auf Touren in den Kalkalpen immer gelungen APRS-Meldungen im Gipfelbereich abzusetzen. Im Funkschatten von Bergen verlief das nicht so verlässlich. Das ist aber kein Problem der Kombination Tracker und Handfunkgerät sondern ist auf den dünnen Ausbau der sonstige APRS Infrastruktur zurückzuführen.

Dieses Selbstbauprojekt ist sowohl technisch als auch finanziell überschaubar. Ich hoffe, dass genaue Positionsmeldungen während meiner Bergtouren die SOTA Jäger unterstützen und gegebenenfalls zu meiner Sicherheit in den Bergen beitragen! Danke an OE5DXL für die Entwicklung der Basis für dieses nette Selbstbauprojekt!

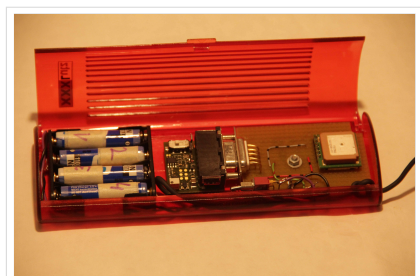
73 Heinz, OE5EEP



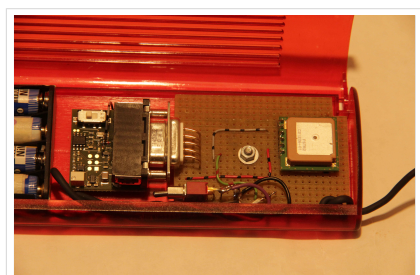
Gesamtansicht



Gesamtansicht



Innenleben



Innenleben-Detail-Tracker+GPS-RX

Prämiert

Der DXL APRStracker wurde mit dem € 2000,- dotierten ÖVSV Innovationspreis 2013 prämiert !
GRATULATION !!

LoRa Variante

Von OE5DXL gibt es auch eine LoRa Variante für Hardware-Tracker des Typs TTGO bzw. LILYGO inkl. Unterstützung für komprimierte Aussendung und Verwendung von MIC-E.

Zu finden ist die fertige INO-Datei in verschiedenen Ausführung, neben anderer Software, wie immer im [HAMNET Indexserver von OE5DXL](#).

Weitere Projekte

Weiter passende Projekte von OE5DXL aus dieser Serie sind

- [DXL - APRSmap](#) - Der neue APRS Client für Windows, Linux und ARM auf OSM Basis
- [TCE Tynocore Linux Projekt](#) - Der mächtige und innovative Digipeater für APRS, Packet Radio, Echolink, u.v.m.

Dateiversionen

Klicken Sie auf einen Zeitpunkt, um diese Version zu laden.

	Version vom	Maße	Benutzer	Kommentar
aktuell	21:47, 14. Sep. 2013	(9 KB)	OE5HPM (Diskussion)	Kompilat vom DXL aprsTracker für Tiny13 mit 10 MHz Systemtakt und 9600 Baud vom GPS-Empfänger, TX ist 1200Bd AFSK.

Sie können diese Datei nicht überschreiben.

Dateiverwendung

Die folgende Seite verwendet diese Datei:

- [DXL - APRStracker](#)

DXL - APRStracker: Unterschied zwischen den Versionen

Versionsgeschichte interaktiv durchsuchen

VisuellWikitext

Version vom 24. September 2023, 20:11

Uhr (Quelltext anzeigen)

OE2WAO ([Diskussion](#) | [Beiträge](#))

K

Markierung: 2017-Quelltext-Bearbeitung

← Zum vorherigen Versionsunterschied

Version vom 24. September 2023, 20:13

Uhr (Quelltext anzeigen)

OE2WAO ([Diskussion](#) | [Beiträge](#))

K

Markierung: 2017-Quelltext-Bearbeitung

Zum nächsten Versionsunterschied →

Zeile 32:

|4800

|1200

– [[Datei:**dxl**-aprsTracker-tiny13-10MHz-4800Bd.zip|verweis=Special:FilePath/dxl-aprsTracker-tiny13-10MHz-4800Bd.zip]]

|-

|ATMEL Tiny13

Zeile 32:

|4800

|1200

+ [[Datei:**Dxl**-aprsTracker-tiny13-10MHz-4800Bd.zip|verweis=Special:FilePath/dxl-aprsTracker-tiny13-10MHz-4800Bd.zip]]

|-

|ATMEL Tiny13

Zeile 38:

|9600

|1200

– [[Datei:**dxl**-aprsTracker-tiny13-10MHz-9600Bd.zip|verweis=Special:FilePath/dxl-aprsTracker-tiny13-10MHz-9600Bd.zip]]

|-

|}

Zeile 38:

|9600

|1200

+ [[Datei:**Dxl**-aprsTracker-tiny13-10MHz-9600Bd.zip|verweis=Special:FilePath/dxl-aprsTracker-tiny13-10MHz-9600Bd.zip]]

|-

|}

Zeile 122:

[[Datei:Oe5eep_2.jpg|rechts|mini|hochkant|200px|Gesamtansicht|verweis=Special:FilePath/oe5eep_2.jpg]]

– [[Datei:**oe5eep_1**.jpg|rechts|mini|200px|Gesamtansicht|verweis=Special:FilePath/oe5eep_1.jpg]]

Zeile 122:

[[Datei:Oe5eep_2.jpg|rechts|mini|hochkant|200px|Gesamtansicht|verweis=Special:FilePath/oe5eep_2.jpg]]

+ [[Datei:**Oe5eep_1**.jpg|rechts|mini|200px|Gesamtansicht|verweis=Special:FilePath/oe5eep_1.jpg]]

-	[[Datei:oe5eep_3.jpg rechts mini 200px Innenleben verweis=Special:FilePath/oe5eep_3.jpg]]	+	[[Datei:Oe5eep_3.jpg rechts mini 200px Innenleben verweis=Special:FilePath/oe5eep_3.jpg]]
-	[[Datei:oe5eep_4h.jpg rechts mini 200px Innenleben-Detail-Tracker+GPS-RX verweis=Special:FilePath/oe5eep_4h.jpg]]	+	[[Datei:Oe5eep_4h.jpg rechts mini 200px Innenleben-Detail-Tracker+GPS-RX verweis=Special:FilePath/oe5eep_4h.jpg]]

Version vom 24. September 2023, 20:13 Uhr

Inhaltsverzeichnis

1	Einleitung	60
2	Software	60
3	Hardware	61
3.1	Trackerschaltung	61
3.2	Geeignete GPS-Empfänger	61
4	Konfiguration	61
5	Referenzprojekte	62
5.1	OE5EEP, Heinz	62
6	Prämiert	63
7	LoRa Variante	64
8	Weitere Projekte	64

Einleitung

Der APRStracker von OE5DXL erlaubt es mit minimalstem Hardwareaufwand in der Betriebsart APRS QRV zu werden. Außerdem wurde bei der Entwicklung der Software darauf geachtet, die derzeitigen Möglichkeiten des APRS-Protokolls in Form von Mic-e optimalst auszunutzen. In der Praxis ist dies durch extrem kurze Frames erkennbar, was im Mobilbetrieb (QSB) erhebliche Vorteile bringt. Zu dem wird das schon etwas in die Jahre gekommene aber durchaus am effektivsten SSID-Pathrouting (im Configtool 'COMPRESSED' genannt) unterstützt.

Als Minimum an Hardware wird benötigt

- ein Mikrocontroller ATtiny13, ATtiny2313, Atmega88 oder auch andere Typen mit (Quarz)-Takt durch Anpassen der I/O Pins im Source
- Quarz frei wählbar ca. 6..20MHz (Im Source eingeben).
- PTT-Transistor
- RC-Tiefpass zum wegfiltern der PWM-Frequenz benötigt.

Software

Opensource Software von OE5DXL, in Assembler geschrieben, welche den Tracker zum Leben erweckt:

Datei:[AprsTracker.zip](#)

Der Assemblercode wird im einfachsten Fall mit dem Compiler 'gavrasm' [Gerd's AVR Assembler](#), welcher sowohl für Linux als auch für Windows verfügbar ist, kompiliert und anschließend in den µC gebrannt (z.B.: mit AVRdude [\[1\]](#) und dem USBasp Programmer [\[2\]](#)). Es ist auch möglich mit dem von ATMEL angebotenen AVR-Studio den Code zu übersetzen.

Fertige Kompilate:

Prozessor	Systemtakt	GPS-Baud	AFSK Baud	Download
ATMEL Tiny13	10 Mhz	4800	1200	Datei:Dxl-aprsTracker-tiny13-10MHz-4800Bd.zip
ATMEL Tiny13	10 Mhz	9600	1200	Datei:Dxl-aprsTracker-tiny13-10MHz-9600Bd.zip

Als Input dienen serielle GPS-Daten (GPRMC und GPGLL), der Tracker generiert daraus anhand der programmierten Konfiguration als Output (nebst PTT) die AFSK-Modulation im APRS-Mic-e Format mit Position, Geschwindigkeit, Fahrtrichtung und Höhe, welche dem Funkgerät zugeführt wird.

Dabei sind (pro Profil) 2 Bakenzeiten und eine Geschwindigkeit einstellbar unter/ueber der langsam/schnell gebakt wird.

Seite 61 von 64

AFSK-Baud und Tonfrequenzen (Shift) sind ebenso frei Konfigurierbar, gebräuchlich sind:

- 300 auf Kurzwelle
- 1200 auf UKW

Zur Einstellung von HUB bzw. Mikrofonpegel eignet sich am besten ein Poti.

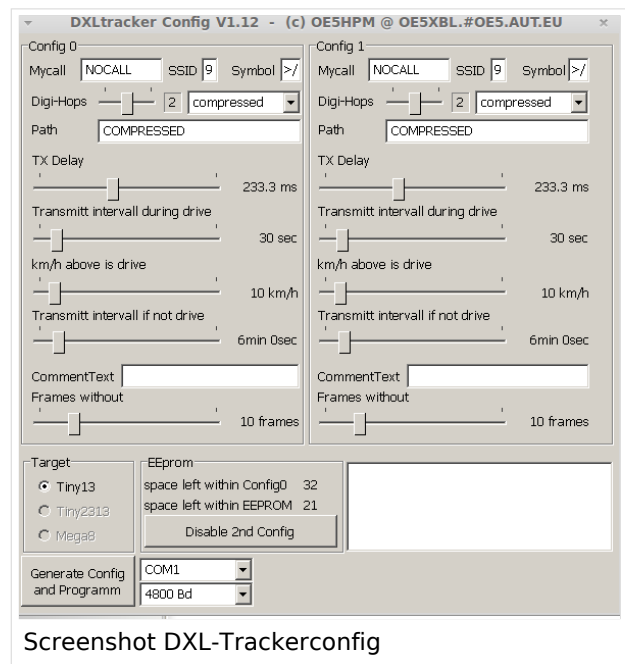
Um die PTT von Handfunkgeräten aufzutasten muss der Mikrofoneingang mit dem PTT Transistor kombiniert werden. Einfachste Variante ist im obigen Schaltungsbild sichtbar, der FET T402 zieht über einen 2k2 Widerstand den MIC-Eingang vom Handfunk gegen Masse und aktiviert dadurch die PTT, über C403 (100nF) wird die Modulation "eingekoppelt".

Als Option kann mit einem Jumper/Schalter (im Schaltbild SW401) zwischen 2 (am Tiny13) oder 4 (auf größeren Prozessoren) User-Profilen ausgewählt werden, zB. Fahrrad/Auto.

An einer Blink-Led für korrekten GPS-Empfang am Prozessor Pin wird gearbeitet.

Zur Konfiguration kommt ein kleines Tool von OE5HPM - DXLtrackerConfig zum Einsatz, dies generiert einen Konfigurationsstring für den DXLtracker und schickt selbigen über die serielle Schnittstelle raus.

[Datei:DXLtrackerConfig.zip](#)



Screenshot DXL-Trackerconfig

Referenzprojekte

OE5EEP, Heinz

Mein Aufbau eines DXL Modems mit einem Fastrax UP501 GPS Empfänger, den ich für Bergtouren im Rahmen des SOTA-Programms einsetzen möchte: Das DXL Modem wurde mir freundlicherweise vom ADL501 zur Verfügung gestellt, der UP501 ist über e-Bay zu beziehen. Als Spannungsquelle für beide Baugruppen gemeinsam dienen 4 NiMH AAA-Zellen von Conrad mit einer nominalen Kapazität von 1100mAh, was für einen ganzen Bergtag leicht ausreicht. Die stabilisierte Spannung von 3,3V für des GPS Modul kann nach Brückung einer Diode an Pin 7 des Sub-D Anschlusses des DXL Trackers abgegriffen werden.

Da ich der Verlässlichkeit von freien Verkabelungen nicht traue, hab ich das GPS Modul mittels der mitgelieferten Steckerleiste auf einer Lochrasterplatte aufgelötet und die Verkabelungen zum DXL Modem auf beiden Seiten der Lochrasterplatte auf eine 9-polige Sub-D Buchse geführt. Dazu kommt noch ein Aus/Ein Schalter und ein Halter für die NiMH Zellen. Eingebaut hab ich das Ganze in eine ehemalige Präsentationsverpackung von 2 Kugelschreibern. Dieses transparente Kunststoffgehäuse erlaubt es, das Aufleuchten der LED im Sendefall von außen zu erkennen und

passt von der Größe her gut zum verwendeten Handfunkgerät (siehe Fotos). Der Aufbau wird am Handfunkgerät einfach mit Gummibändern (Stücke von einem Fahrradschlauch) befestigt. Die einzige elektrische Verbindung zum Handfunkgerät ist eine Audioleitung zur Mikrofonbuchse, in meinem Fall mit einem 2,5mm Mono-Klinkenstecker.

Bisher hab ich den Tracker auf mehreren Bergtouren eingesetzt und positive Erfahrungen gemacht. Der GPS Empfänger findet binnen weniger Minuten eine Position. Die Empfindlichkeit ist ausgezeichnet. Teilweise kann ich im Inneren von Gebäuden eine Position bekommen, allerdings dann mit größerem Fehler. Jedenfalls ist der GPS-Empfänger deutlich empfindlicher als der in meinem LG Smartphone. Im Zusammenspiel mit einem schon etwas älteren IC-2e mit etwa 3W Sendeleistung und Gummiantenne ist es mir auf Touren in den Kalkalpen immer gelungen APRS-Meldungen im Gipfelbereich abzusetzen. Im Funkschatten von Bergen verlief das nicht so verlässlich. Das ist aber kein Problem der Kombination Tracker und Handfunkgerät sondern ist auf den dünnen Ausbau der sonstige APRS Infrastruktur zurückzuführen.

Dieses Selbstbauprojekt ist sowohl technisch als auch finanziell überschaubar. Ich hoffe, dass genaue Positionsmeldungen während meiner Bergtouren die SOTA Jäger unterstützen und gegebenenfalls zu meiner Sicherheit in den Bergen beitragen! Danke an OE5DXL für die Entwicklung der Basis für dieses nette Selbstbauprojekt!

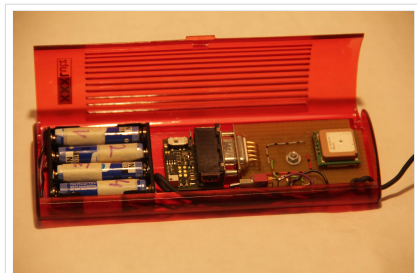
73 Heinz, OE5EEP



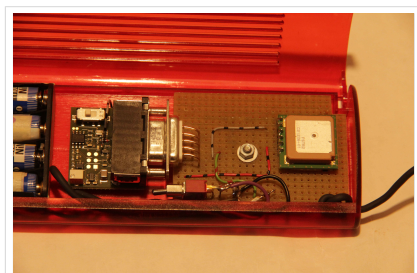
Gesamtansicht



Gesamtansicht



Innenleben



Innenleben-Detail-Tracker+GPS-RX

Prämiert

Der DXL APRStracker wurde mit dem € 2000,- dotierten ÖVSV Innovationspreis 2013 prämiert !
GRATULATION !!

LoRa Variante

Von OE5DXL gibt es auch eine LoRa Variante für Hardware-Tracker des Typs TTGO bzw. LILYGO inkl. Unterstützung für komprimierte Aussendung und Verwendung von MIC-E.

Zu finden ist die fertige INO-Datei in verschiedenen Ausführung, neben anderer Software, wie immer im [HAMNET Indexserver von OE5DXL](#).

Weitere Projekte

Weiter passende Projekte von OE5DXL aus dieser Serie sind

- [DXL - APRSmap](#) - Der neue APRS Client für Windows, Linux und ARM auf OSM Basis
- [TCE Tinycore Linux Projekt](#) - Der mächtige und innovative Digipeater für APRS, Packet Radio, Echolink, u.v.m.