

WSPR einfach

***Versuche mit Degen DE 1103 und
QRP Labs Arduino Shield***

Christoph Schwärzler, OE1CGS

Februar 2016

Abstract

This document deals with a simple WSPR-Station which is easy to build and run. The transmitter is a 0,2 W kit by G0UPL's QRP labs¹ and the receiver a HF-radio Degen DE 1103. Receiving antenna is the simple wire antenna provided with the radio and transmitting is via a matched, endfed inverted-L wire antenna. A manual QRP tuner, sold as a kit by Pacific Antenna (formerly Hendricks QRP kits)² is used for matching the high impedance of this antenna. With this simple transmitter confirmed QSO's of distances up to 18161 km (OE to ZL) were achieved.

Zusammenfassung

In diesem Dokument wird eine mit minimalem Aufwand erstellte und betriebene WSPR-Station beschrieben. Als Sender dient ein 0,2 W starker Bausatz von G0UPL's QRP Labs¹ und als Empfänger ein Kurzwellenradio Degen DE 1103. Während als Empfangsantenne der mit dem Radio mitgelieferte Draht dient, wird der Sender an eine angepasste, endgespeiste inverted-L Drahtantenne angeschlossen. Die Anpassung erfolgt über einen manuellen Tuner von Pacific Antenna (vormals Hendricks QRP kits)². Mit dieser Ausrüstung konnten bestätigte QSO's bis zu einer Distanz von 18161 km (OE - ZL) geführt werden.

1.) Sender

Der Sender ist der „Arduino Shield“ Bausatz von QRP Labs zusammen mit dem „Si5351A synthesizer“ und dem passenden „Low Pass Filter“. Dieses Shield kann auf einen Mikrocontroller „Arduino Uno“ direkt aufgesteckt werden. Auf den aufwändigeren „OCXO/Si5351A“ Bausatz für das DDS-Modul wurde bewusst verzichtet, dafür jedoch auf eine möglichst frequenzstabile Realisierung des Si5351A-Moduls geachtet. Dazu wurde ein Stück eines metallenen Abstandhalters mittels eines Tropfens schnellhärtendem 2-Komponenten-Kleber auf dem Gehäuse des 27 MHz - Quarzes befestigt (Erhöhung der thermischen Trägheit) und das gesamte Modul mit einem kleinen Gehäuse aus Polystyrol-Hartschaum („Styrofoam“) thermisch isoliert. Außerdem ist im Programm integriert, dass der Sender durch einen „Park mode“ in den Sendepausen warmgehalten wird (siehe weiter unten).

Auf dem Arduino Shield befindet sich ein kleiner Sendeverstärker in Form eines MOSFET der Type BS170. Auf dem Shield ist auch Platz für zwei weitere BS170 vorgesehen. Um die damit mögliche, höhere Ausgangsleistung zu erreichen, müsste aber

auch mit einer zusätzlichen externen Spannungsversorgung gearbeitet werden. Mit einem BS170 und der Versorgung mit 5V durch die Arduino-Platine beträgt die Sendeleistung in etwa 200 mW.

Für jedes Frequenzband ist ein eigenes 7-Element Tiefpassfilter erforderlich. Es muss jeweils das passende in den Sendeausgang des Arduino Shields gesteckt werden. Die bei diesen Experimenten benutzten WSPR-Bänder sind:

40 m: 7.040.000 – 7.040.200 Hz
30 m: 10.140.100 – 10.140.300 Hz
20 m: 14.040.000 – 14.040.200 Hz

Das Programm zur Steuerung des DDS und der WSPR-Übertragung ist im Anhang beschrieben und zum Download verfügbar. Es ist sehr einfach gehalten und verzichtet z.B. auf eine exakte Zeitsteuerung für den Beginn der Aussendungen, welche mit geraden UTC-Minuten definiert sind. Ebenso ist keine dynamische Erstellung des WSPR-codes („Channels“) vorgesehen. Die Liste der Kanäle muss vorab extern generiert³ und im Programmcode direkt integriert werden.

Ich habe sämtliche Komponenten des Senders in ein 3D-gedrucktes PLA-Gehäuse⁴ eingebaut (Bild 1).

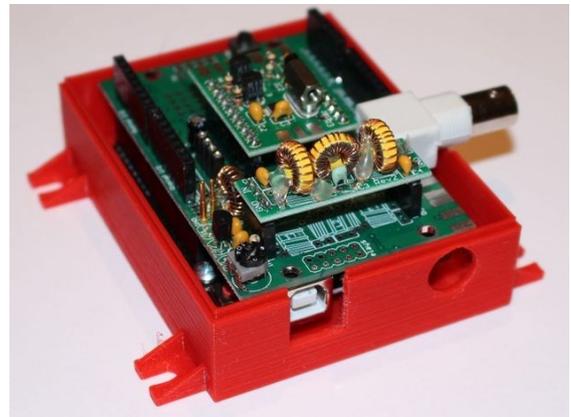


Bild 1: WSPR-Sender im 3D-gedruckten Gehäuse (hier ohne Deckel abgebildet)

2.) Empfänger

Der Empfänger besteht aus einem handelsüblichen Weltempfänger der Marke Degen DE1103 mit der mitgelieferten einfachen Drahtantenne. Er dient vor allem zur Kontrolle der eigenen Aussendung, aber es konnten mit ihm auch mehrere fremde WSPR Sprüche erfolgreich empfangen werden. Bedingt durch den Aufstellungsort im Keller eines Einfamilienhauses und die Antennenführung war nur ein sehr schlechtes Signal-Rausch-Verhältnis (SNR) zu erzielen. Daher wurde auf die Einspielung dieser nicht aussagekräftigen Empfangsreporte in das

internationale WSPR-Netz⁵ – mit Ausnahme einzelner Tests – verzichtet. Zur Kontrolle der eigenen Aussendung jedoch war dieser RX mehr als ausreichend, typische SNR der eigenen Aussendung betragen +9 bis +12 dB⁶.

Etwas trickreich gestaltet sich die Frequenzeinstellung am Radio, da der Frequenzbereich von WSPR nur 200 Hz pro Band beträgt. WSPR-Aussendungen werden im USB mit einer Ablage von 1500 Hz übertragen, daher muss das Radio sehr genau auf eine Frequenz von 1500 Hz unter der jeweiligen Bandmitte eingestellt werden.

Die Mitten dieser WSPR-Frequenzbereiche in den Bändern von 20, 30 und 40 m liegen bei:

40 m: 7040,2 kHz
30 m: 10140,3 kHz
20 m: 14097,2 kHz

Ich habe auf dem Einstellrad für die Ablage (SSB Fine Tune Knob) des Radios drei Farbmarkierungen vorgenommen (Bild 2). Ein weißer Strich markiert eine Ablage von 0 Hz, also diejenige Einstellung bei der exakt die digital angezeigte Frequenz gewählt ist.



Bild 2: Farbmarkierungen auf dem Einstellrad für die Frequenzablage erleichtern die Frequenzeinstellung

Ein grüner Strich kennzeichnet eine negative Ablage von genau 1000 Hz (typisch bei USB-Betrieb) und ein roter Strich eine positive Ablage von 1000 Hz (typisch LSB-Betrieb). Mit diesen Markierungen und fällt es relativ leicht, sogar die schmalen Frequenzbänder der WSPR-Betriebsart rasch aufzufinden.

Die exakten Positionen zur Erstellung dieser Markierungen sind durch Empfang eines Senders mit bekannter Frequenz (in Europa z.B. der russische Zeitzeichensender RWM auf 9996 kHz) bestimmbar. Dazu muss das Radio über ein 3,5 mm Klinkenkabel und idealerweise einem Interface⁷ mit der

Soundkarte eines Computer verbunden werden auf welchem eine geeignete Software wie z.B. Argo⁸ installiert ist. Mit dieser Software kann dann das NF-Signal auf wenige Hertz genau gemessen und daraus die eingestellte Ablage ermittelt werden.

Um nun die für WSPR erforderliche Ablage von 1500 Hz zu erhalten, sind für die Bandmitten (siehe oben) folgende Einstellungen am Radio vorzunehmen:

Frequenzeinstellungen am Radio:

40 m:	7039 kHz	USB 300 Hz
30 m:	10139 kHz	USB 200 Hz
20 m:	14096 kHz	USB 300 Hz

3.) Handhabung des Systems

Im ersten Schritt muss der Programmcode an den beabsichtigten Betrieb angepasst werden (siehe auch Anhang). Dies erfolgt über die Arduino IDE. Nachdem das Programm auf den Mikrocontroller aufgespielt wurde, wird das Shield von QRP Labs aufgesteckt. Nun muss das zur Sendefrequenz passende Tiefpassfilter und eine Dummyload am Sendausgang angeschlossen werden. Der Trimmer auf dem Shield zur Biaseinstellung des Sendetransistors muss nun gegen den Uhrzeigersinn bis zum Anschlag gedreht werden. Damit wird der Bias und somit die Sendeleistung minimal und der Arduino Uno samt Shield an den USB-Port des Computers angeschlossen. Alternativ ist auch eine Versorgung durch eine Batterie oder einen Akkumulator (eigen Buchse auf der Platine des Arduino Uno) möglich.

Nach dem der Versorgungsspannung beginnt sofort der Sendebetrieb. Nun kann mit dem Empfänger, welcher über ein Interface mit der Soundkarte eines Computers verbunden ist die korrekte Aussendung der WSPR-Message überprüft werden. Dazu muss auf dem Computer die Software „WSPR“⁹ installiert sein.

Unter Zuhilfenahme eines geeigneten Messgeräts (QRP-Leistungsmesser oder Oszilloskop mit Dummyload) muss nun mittels vorsichtiger Drehung des Trimmers auf dem Arduino Shield im Uhrzeigersinn die gewünschte Sendeleistung eingestellt werden.

Wird nun anstelle des Dummyloads eine geeignete Antenne angeschlossen, so ist der simple WSPR-Sender fertig. Es macht Spaß zu sehen, wie mit geringster Sendeleistung damit ganze Kontinente überbrückt werden können. Gut bewährt hat sich dabei eine endgespeiste, invertierte L-Antenne,

welch mit einem manuellen Tuner SLT+ von Pacific Antenna angepasst wurde.

Umgekehrt ist es bereits mit einem einfachen Kurzwellenempfänger möglich, selbst weit entfernte und mit geringer Leistung sendende WSPR-Baken zu empfangen und automatisiert auszuwerten. Dazu genügt im einfachsten Fall eine ca. 10m lange Drahtantenne am Antenneneingang des Radios. Natürlich ist eine Empfangsverbesserung durch optimierte Antennen möglich¹⁰.

Anhang: Programmcode

Die Software¹¹ zur Steuerung des DDS und Ausgabe der WSPR-Meldung ist auf der Arduino IDE (Version 1.6.5) entwickelt worden. In der Hauptschleife wird zyklisch (z.B. alle 2 Minuten) eine Übertragung gestartet. Der Inhalt der WSPR-Message wird zu Beginn des Programmcodes dem Variablen-Array „message“ zugewiesen und muss bereits als Folge der zu sendenden Channels vorliegen. Diese Message muss u.a. dann geändert werden, wenn mit einer anderen Sendeleistung gearbeitet wird.

Neben eine Anpassung der WSPR-Message im Programmcode sind auch noch folgende Einstellungen zu beachten:

- Sendefrequenz (in Hz). Diese muss innerhalb der WSPR-Bandbreiten (s.o.) gewählt werden. Eine etwaige idiosynkratische bzw. Temperaturkompensation des Quarzes ist dabei nicht hier, sondern über die Definition von F_XTAL vorzunehmen.
- Sendeintervall (in ms). Dieses muss mindestens 2 Minuten (rund 120.000 ms) betragen, kann aber auch ein ganzzahliges Vielfaches davon sein. Ist ein längerer, autonomer Sendebetrieb vorgesehen, so sollte der Temperatureffekt auf die Frequenz des Taktquarzes über diese Größe zu gut wie möglich kompensiert werden.

Der Programmcode beginnt ca. 1 Sekunde nach Anlegen der Betriebsspannung mit der Aussendung der ersten Bake. Daher ist es unbedingt erforderlich, die Spannung händisch zur 59. Sekunde einer ungeraden Minute (z. B. 13:01:59) anzulegen. Ab dann kann das System bei vernünftiger Kalibrierung (siehe „Sendeintervall“ oben) durchaus einige Stunden ohne weiteren Eingriff betrieben werden.

¹ <http://www.qrp-labs.com/>

² <http://www.qrpkits.com/sltplus.html>

³ <http://physics.princeton.edu/pulsar/K1JT/WSPRcode.exe>

Leider ist dieses Programm nur bis zu Windows XP lauffähig.

⁴ <http://www.thingiverse.com/thing:1389636>

⁵ <http://wsprnet.org>

⁶ Dieser Artikel folgt der SNR-Definition des WSPR-Modus. Alle Angaben beziehen sich daher auf eine Bandbreite von 2500 Hz. Zur Umrechnung auf eine Bandbreite von 1 Hz (VOACAP-Definition) muss daher zum angegebenen SNR ein Wert von 34 dB addiert werden.

⁷ Im WWW sind unzählige Schaltungen dazu zu finden. Ich habe eine bereits vorhandenes Interface für den Degen zu einem Tablet (<http://www.oe1cgs.at/mikrofonadapter/>) zusammen mit einem Adapterkabel von Tablet (4-pol. Stecker 3,5 mm) auf 3,5 mm Stereoklinke benutzt.

⁸ <http://www.weaksignals.com/>

⁹ <http://physics.princeton.edu/pulsar/k1jt/wspr.html>

¹⁰ Antennen mit einer Fußpunktimpedanz von 50 Ω sollten idealerweise mit einem passenden Anpassungsübertrager an einen Weltempfänger angeschlossen werden. Siehe dazu auch : <http://www.oe1cgs.at/eingangsimpedanz-degen-de-1103/>

¹¹ <http://www.oe1cgs.at/downloads/>