

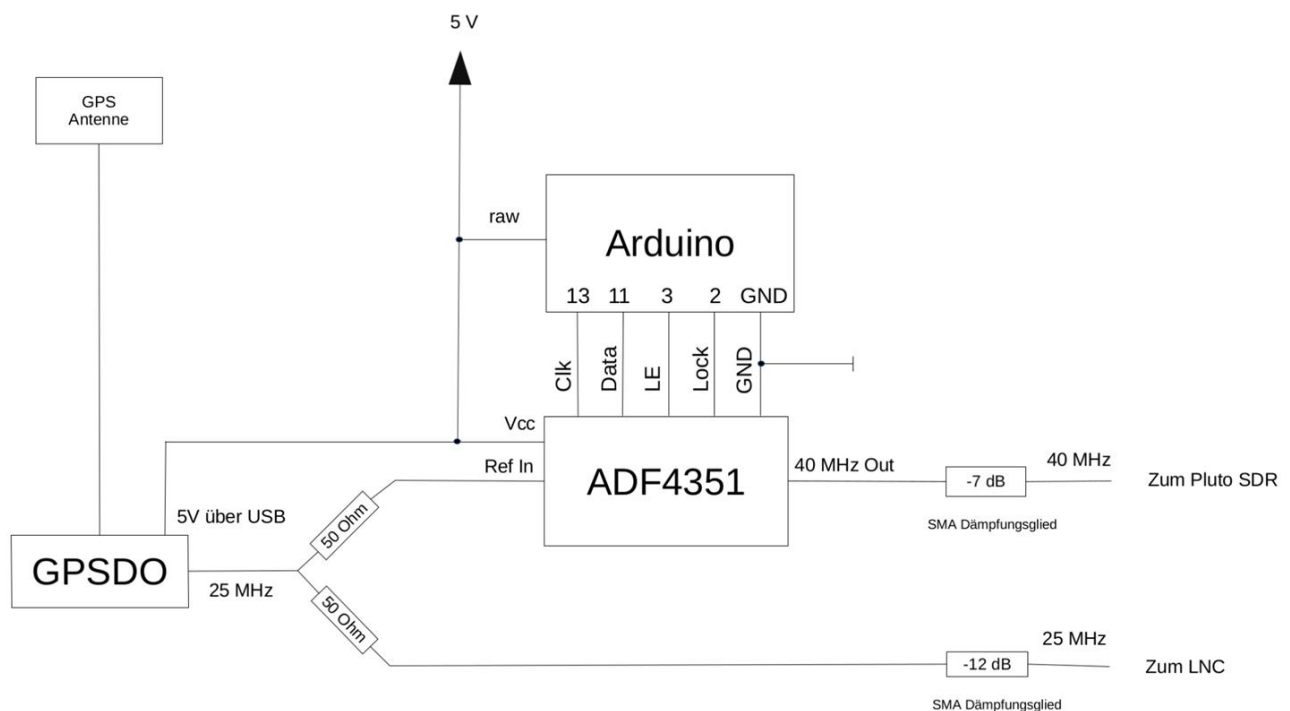
GPSDO Stabilisierung für 25MHz LNBs und Adalm Pluto SDR

20230823 – OE5RNL

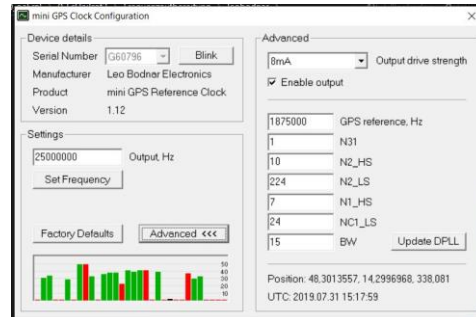
Die Frequenzstabilität spielt bei einer QO100 TRX Anlage, selbst bei SSB, eine große Rolle. Weder die verwendeten SAT-LNBs noch die meisten SDRs verfügen über eine wirklich ausreichende Frequenzstabilität, die über ein QSO hinausgeht...

Mit der SDR Console von Simon Brown kann der RX mit der Bake des QO100 synchronisiert werden. Will man jedoch auch eine exakte TX Frequenz haben und andere SDR Programme als die SDR Console verwenden (z.B.: SDR Angel), so ist eine externe Synchronisation unumgänglich.

Beschrieben wird hier die Stabilisierung eines umgebauten 25 MHz LNBs und eines Adalm Pluto, der eine 40 MHz Referenz benötigt, mit einem GPS disziplinierten Oszillator (GPSDO) und einem ADF4351 PLL-Oszillator.



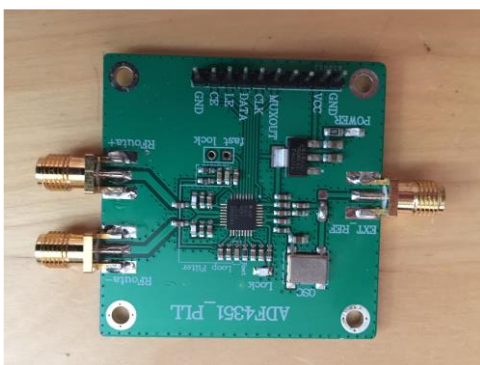
Ich verwende das GPSDO Modul „Mini Precision GPS Reference Clock“ der Firma Leo Bodnar. Dieses Gerät erzeugt an einem Ausgang eine Frequenz, die sich zwischen 400 Hz und 800 MHz über ein Windows Programm per USB einstellen lässt. Es gibt im Internet aber auch 25 MHz TCXOs mit hoher Genauigkeit die alternativ eingesetzt werden können.



Für unseren Zweck habe ich die Ausgabefrequenz des GPSDO auf 25 MHz und die Ausgangsstärke auf 8 mA eingestellt.

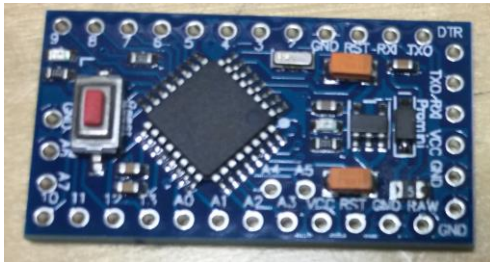
Das 25 MHz Signal wird über einen einfachen Leistungsteiler aus zwei 50 Ohm Widerständen gesplittet. Der Pegel eines Ausgangs des Splitters wird über ein 12 dB Dämpfungsglied reduziert und an den umgebauten LNB weitergeleitet. Grundsätzlich kann die Konfiguration auch für 27 MHz LNBs verwendet werden. Dabei ist jedoch die Ladesoftware im Arduino anzupassen da die Referenzfrequenz für die 40 MHz PLL dann 27 MHz beträgt. Mehr dazu weiter unten.

Für die Erzeugung des 40 MHz Signals für den Pluto SDR verwende ich den Baustein ADF4351 von Analog Devices der eine PLL mit einem Oszillator vereint. Er kann Referenzfrequenzen zwischen 10 und 250 MHz verarbeiten und erzeugt ein Ausgangssignal im Bereich von 35 MHz und 4,4 GHz. Im Internet gibt es die fertige Platine die den ADF4351, einen Stabi IC, das Loop Filter und einen 10 MHz Referenzoszillator beinhaltet. Der eingebaute 10 MHz Oszillator lässt sich für unsere Anwendung über eine Lötbrücke deaktivieren. Wer auf ganz sicher gehen will kann den Oszillator auch auslöten. Die Platinen kosten zwischen 14 und 24 Euro.



Der Zweite Ausgang des Splitters wird an den Referenzeingang des ADF4351 angelegt. Der Baustein erzeugt mit der verwendeten Software, die für den Pluto SDR benötigten, 40 MHz. Vom Gegentaktausgang des ADF4351 wird nur ein Port genutzt. Die Ausgangsleistung ist zu hoch und wird mit einem Dämpfungsglied um 7 dB abgesenkt. Alternativ kann man den Leistungspegel auch in der Software über die Funktion `SetPowerADF4351()` einstellen.

Das Laden der Register des ADF4351 erfolgt über einen Arduino Pro Mini. ACHTUNG: die 3,3V Version verwenden!!! Beim Anlegen der Spannung wird nach einer Zeit von einer Sekunde der ADF4351 Baustein programmiert. Dann legt sich der Arduino schlafen und sollte so keine Störungen verursachen.



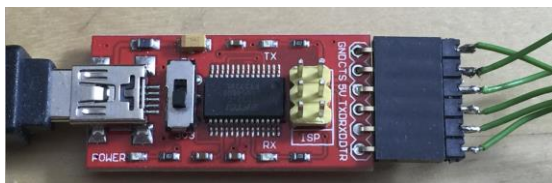
Als Grundlage habe ich die geniale Software von Matthias Busse verwendet. In der Originalsoftware erfolgt die Frequenzeingabe über die serielle Schnittstelle, ohne dass die Werte dauerhaft gespeichert werden. Es müsste also bei jedem Neustart die Programmierung über die serielle Schnittstelle erfolgen.

Ich habe mich daher für die fixe Programmierung der Ausgangsfrequenz entschieden. Die Frequenz wird über die Funktion `SetFrequencyADF4351(40.0)` im Klartext eingestellt. Damit muss man sich nicht mit den „echten“ Registerwerten herumplagen. Die Referenzfrequenz ist in der Variablen „quarzFreq“ mit dem Wert 25.0 hinterlegt, oder eben 27 MHz bei 27 MHz LNBs. Die Funktionalität über die serielle Schnittstelle des Originalprogramms bleibt für eventuelle Tests erhalten, wird aber hier nicht benötigt.

Zum Programmieren des Arduino verwende ich die Arduino GUI. Damit können also auch gleich eventuell notwendige Anpassungen vorgenommen werden.

Die leicht angepasste Software von Matthias Busse ist als Download im Anhang des Artikels auf der Technikseite des ÖVSV verfügbar. Herzlichen Dank an Matthias Busse für die Genehmigung zur Weiterverwendung.

Zu beachten ist, dass die serielle Schnittstelle des Arduino Pro Mini nur 3,3V verarbeitet. Zum Programmieren ist also ein externes Interface notwendig.



Benötigt werden die Signals GND, 5V, TRX, RXD und DTR

Einstellungen in der Arduino IDE Menüpunkt „Werkzeuge“:

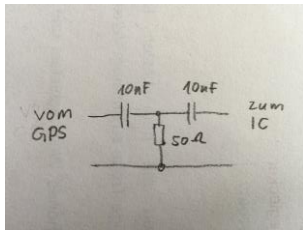
Board: Arduino Pro or Pro Mini
Prozessor: ATMegae328P (3,3V, 8MHz)
Programmer: Arduino as ISP

Alternativ kann man natürlich auch einen Arduino mit eingebauter USB-Serial-Schnittstelle verwenden.

Im github ist nun nur mehr die Version für NS VS Code mit plattformio hinterlegt. Am PC habe ich MS VS Code und die Arduioni IDE installiert.

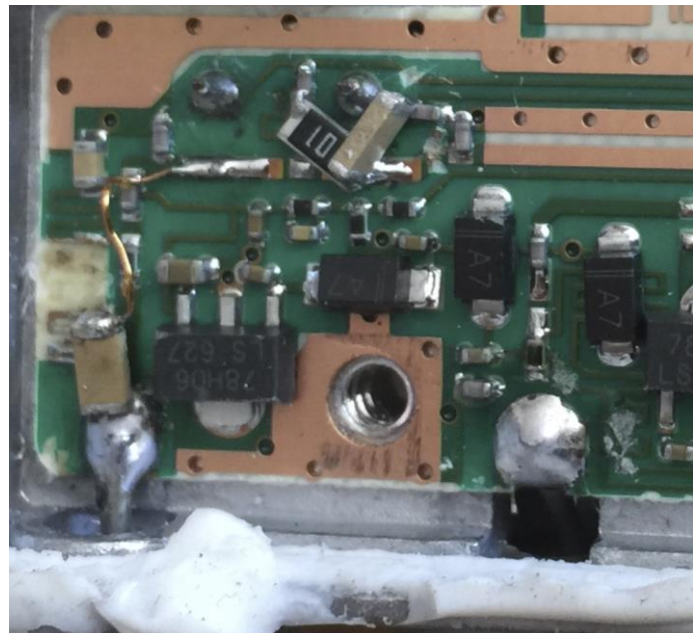
Link: <https://github.com/oe5rnl/ADF4351-QO100.git>

Umbau LNB:



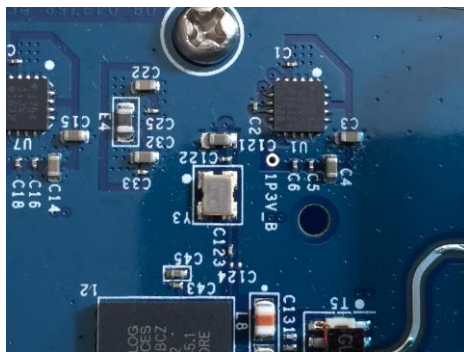
Eine genaue Beschreibung für den Umbau des Octagon Green 4 Port mit Fotos gibt es auch der Seite:
https://wiki.batc.org.uk/Es%27hail-2_LNBs_and_Antennas

„vom GPS“ ist der Ausgang des AFD4351 nach dem 12 dB Dämpfungsglied. Im LNB also die linke Buchse im Foto unten. Dieser Eingang wird für die Synchronisation verwendet. „zum IC“ ist der Punkt, an dem der Quarz im LNB mit dem IC verbunden war.



Oben die Übersicht. Rechts die Details.
Ginge auch etwas schöner :-)

Umbau Pluto SDR:



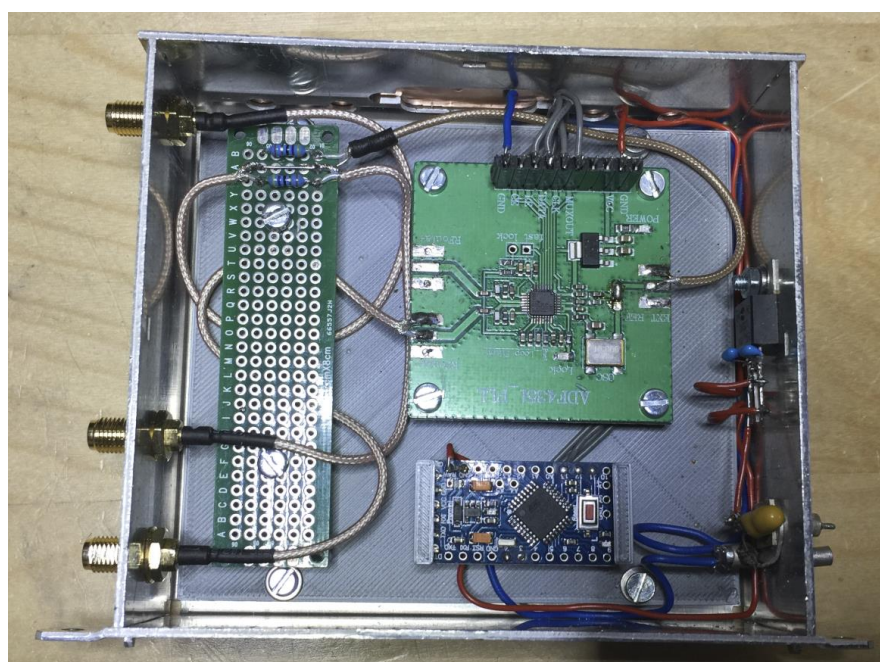
Den TCXO Y3 auslöten. Das ist der silberne Bauteil in der Mitte des Fotos mit den 4 Anschlüssen. Das geht am besten mit Heißluft. An der Verbindung zwischen dem TCXO und C123 wird das Oszillatorsignal vom ADF4351 über das Dämpfungsglied (extern) eingespeist. Ich habe dazu über ein kurzes Koaxkabel eine SMA Buchse zwischen der TX und RX Buchse am Pluto eingebaut.

Die Komponenten habe ich in ein Metallgehäuse eingebaut. Die SMA Buchsen mit Kabel gibt es auch im Internet.

25 MHz vom GPSDO

25 MHz zum LNB

40 MHz zum Pluto



SDR

Auf der rechten Seite befindet sich noch ein 5V Stabi-IC und der 12 V Eingang.

Links:

Die Seite von Matthias Busse enthält auch viele weitere interessante Beiträge.
<http://shelvin.de/den-adf4351-frequenzgenerator-35mhz-bis-44ghz-vom-arduino-uno-ansteuern/>

Die Seite zum ADF4351 von Analog Devices
<https://www.analog.com/en/products/adf4351.html#product-overview>

Die Beschriebene Platine ist bei verschiedenen Händlern im Internet zu haben.

Die Homepage für den PlutoSDR bei Analog Devices
<https://www.analog.com/en/design-center/evaluation-hardware-and-software/evaluation-boards-kits/adalm-pluto.html>

Für den Pluto SDR gibt es im Internet ebenfalls viele Anbieter.

Der Arduino Pro Mini ist auf der Seite <https://store.arduino.cc/arduino-pro-mini> als „retired“ ausgewiesen. Im Internet gibt es jedoch noch viele Bezugsquellen.