

Arduino Antennen Analysator - DG7EAO

Auf der Seite von SierraRadio hatte Beric Dumm - K6BEZ einen Antennenanalysator veröffentlicht. Dieser bestand aus einem preiswerten AD9850 DDS Board (Ebay 8 €), einem Arduino Micro und ein paar Bauteilen für eine SWR Brücke. Im Internet wurde auch der Sourcecode für den Arduino und die PC - Auswertesoftware angeboten. Leider klappte der Nachbau nicht sofort, da die Software ein paar kleinere Bugs enthielt. Es war aber kein großes Problem die Software anzupassen, da sie gut dokumentiert war. Die Oberfläche der PC Software habe ich neu gestaltet und meinen Erfordernissen angepasst.

Die Hardware habe ich ein wenig geändert:

Der ursprüngliche verwendete Arduino Micro verfügt über eine Miniatur USB Schnittstelle. Dieses Modul ist allerdings relativ teuer in der Beschaffung. Deshalb habe ich einen Arduino Mini Pro verwendet. Dieser hat keine USB Schnittstelle. Man bekommt aber für 4.50 € einen USB > RS232 TTL Adapter der gut funktioniert. Hierbei befindet sich die komplette Elektronik in dem USB Stecker. Das AD9850 Modul gibt es bei Ebay in 2 Versionen, diese sind nicht Pin-kompatibel. Ich benutze das blaue Modul (ohne rote LED).

Der Aufbau kann ohne Probleme auf einer Lochrasterplatine erfolgen. Ein Leiterplattenlayout (Eagle) steht auch zur Verfügung.

Die Gesamtkosten wurden mit unter 40 € angegeben. Mit Gehäuse und Leiterplatte wird es wohl etwas mehr werden.

Was kann das Gerät?

Das Arduino - Modul sendet in Kombination mit der PC - Software die Daten für einen Frequenz - Sweep z.B. 1 .. 30 MHz an das DDS - Modul. Der Signalausgang wird mit der Antenne verbunden. Gleichzeitig wird über eine SWR - Brücke das SWR gemessen und über die RS232 Schnittstelle zur PC - Software gesendet. Diese zeichnet eine Frequenzkurve mit den gemessenen SWR - Werten. Mit dieser Messung ist es möglich die Resonanzpunkte von selbstgebauten Antennen zu ermitteln. Die SWR Werte geben einen guten Anhaltspunkt über den Frequenzverlauf. Allerdings sind es keine genauen Absolut Werte. Zum experimentieren mit Antennen reicht die Auflösung aber völlig aus. Später könnte man vielleicht noch die Messkurve der verwendeten Germanium - Dioden linearisieren und dadurch die Genauigkeit der SWR - Werte erhöhen. Auf die Resonanzpunkte hat dies keine Auswirkung.

Ein MiniVNA ist wesentlich teurer und kann zusätzlich Impedanzverläufe und Fußpunktwidestände messen. Dies ist bei dieser einfachen Schaltung nicht möglich.

Stückliste

AD9850 Modul DDS Signal Generator 40Mhz CP13010 D54

Eckstein Komponente 12,45 €

Arduino Pro Mini Atmega328P Board Modul 5V/16MHz AR01011 J32

Eckstein Komponente 6,99 €

USB auf RS232 TTL UART PL2303HX Konverter Adapter Kabel CP12005 F44

Eckstein Komponente 4,95 €

Eckstein Komponente bietet die Module mit kurzer Lieferzeit im Internet an <http://stores.ebay.de/Eckstein-Komponente>

Mit längerer Lieferzeit kann man die Komponenten meist preiswerter direkt aus China beziehen.

R1, R1A, R2, R2A,R3, R3A	100 Ohm 1%, RM 7,5
R4, R8	10 K RM 7,5
R5, R9	100 K RM 7,5
R6, R10	4,99 K RM 7,5
R7, R11	648 R RM 7,5
R20, R21	Drahtbrücke
C1	1u RM 5
C2, C4	100n RM 5
C3, C5, C6	10n RM 5
IC1	LM358
IC - Fassung	8-polig
D1, D2	Germanium Diode AA143
JP1	Stiftleiste 2pol. 2,54 mm Raster (Antenne)
SV1	Buchsenleiste 7pol. 2,54 mm Raster
SV2	Buchsenleiste 7pol. 2,54 mm Raster
U\$1	Buchsenleiste 2 x 12pol. 2,54 mm Raster (Arduino Mini)
Stiftleiste	2 x 12pol. 2,54 mm Raster (Arduino Mini)
Stiftleiste	1 x 4pol. Abgewinkelt 2,54 mm Raster (Arduino Mini + / -/ RX / TX)
Stiftleiste	2 x 7pol. 2,54 mm Raster (AD9850)
Junper	3 Jumper 2,54 mm
Antennenbuchse	PL oder BNC (JP1)

Aufbauhinweise

Die Schaltung lässt sich recht einfach auf einer Lochrasterplatine gemäß Schaltplan aufbauen.

Ich habe versucht auch eine Leiterplatte mit Eagle aufzubauen (noch nicht getestet).

Hierzu ein paar Hinweise:

1. Die Stiftleisten (2 x 12) in das Arduino – Modul einlöten an den beiden Langseiten.
Stifte nach unten !
2. Eine abgewinkelte Stiftleiste an die kleine Seite mit der Beschriftung
TX - RX - +5V - GND anlöten. Hier wird später der USB > RS232 TTL Adapter aufgesteckt.
Der von mir verwendete Adapter hatte schon Steckbuchsen. Die Farben waren
TX Weiss
RX Grün
5V Rot
Gnd Schwarz
Die Spannungsversorgung erfolgt über den USB > RS232 TTL Adapter. Ein zusätzliches
Netzteil ist nicht erforderlich.
3. Auf der Leiterplatte bei „Arduino Mini U\$1“ Buchsenleisten (2x12) einlöten.
Das Arduino Modul wird so aufgesteckt, dass die abgewinkelte Leiste Richtung
JP4/JP5 zeigt. Die Lötunkte von JP4/JP5 können optional als Anschlusspunkt für
die abgewinkelte Leiste (auf dem Modul) genutzt werden.
4. Die Stiftleisten (2x7) bei dem DDS Modul einlöten. Stifte nach unten !
Die Pins D0 .. D6 werden nicht benutzt. Auf dem Modul jeweils einen Jumper setzen bei
J1, J2, J3.
5. Auf der Platine die Buchsenleisten 2 x 7) bei SV1 und SV2 einlöten.
6. Das DDS Modul wird in die Buchsenleisten SV1 und SV2 gesteckt.
Die Pins mit GND – Out – GND zeigen nach oben (SV1).
7. In die Lötunkte bei JP1 kommt eine Stiftleiste mit 2 Pins. Hier wird die Antennebuchse
angeschlossen. Der untere Pin ist Masse.
8. An JP3 sind weitere Pins des Arduino Boards heraus geführt. Diese dienen evtl. späteren
Erweiterungen, z.B. LCD Anzeige und können erst einmal unbestückt bleiben.
9. Alle weiteren Bauteile gemäß Bestückungsplan verlöten. Bei R20 und R21 sind Drahtbrücken
einzusetzen.
10. Die Eagle - Dateien bestehen aus:
VNA1.sch Schaltplan
VNA1.brd Platinenlayout
11. Die Programmierung des Arduino – Boards kann mit dem USB > RS232 Adapter und
der Arduino Programmier IDE erfolgen. Nach dem Start muss der kleine Taster auf
dem Board kurz gedrückt werden. Der Sourcecode befindet sich in der Datei DDS_sweeper.ino.

Software

Die Software ist unter DotNet programmiert und benötigt eine aktuelle Version des DotNet Frameworks (Version 4.0 oder höher). Dieses kann kostenlos von Microsoft herunter geladen werden. Eine Software Installation ist nicht erforderlich. Es reicht das Verzeichnis mit den Programmdateien an eine beliebige Stelle zu kopieren und die Datei **VNA.exe** zu starten.

Nach dem Programmstart wählt man unter Port die Com - Schnittstelle des USB – Adapters aus. Mit einem Click auf **Connect** wird eine RS232 Verbindung zum Arduino Modul hergestellt. Unter **Start MHz** und **Stop MHz** ist der Frequenzbereich 1 .. 30 MHz voreingestellt. Diesen kann man beliebig ändern oder mit der Auswahlbox **Bands** ein Amateurfunkband wählen. Durch einen Click auf **Sweep** werden die Daten an das Arduino Modul gesendet, dieses antwortet mit den entsprechenden SWR - Werten. Aus diesen wird dann eine Verlaufskurve gezeichnet. Unter **Steps** ist als Wert 101 eingestellt. Dies bedeutet, dass bei einem Frequenz - Sweep 101 Frequenzen gemessen werden. Man kann diesen Wert bis auf 1000 erhöhen, dann bekommt man eine höhere Auflösung mit bis zu 1000 Messpunkten – die Messung dauert dann aber etwas länger. Die Auflösung erhöht sich auch, wenn man den Frequenzbereich verkleinert, z.B. statt 1 .. 30 MHz >> 13 .. 15 MHz. Dies erreicht man auch durch Drücken der Tasten **Zoom In** und **Zoom Out** und dann **Sweep**. Die Grafik ist auf einen SWR Bereich von 1 .. 10 voreingestellt. Mit dem Schieberregler **Maximum SWR** kann man den Wert zwischen SWR 2 .. 30 zoomen. Zusätzlich kann man die Grafik auch zoomen, indem man mit der linken Maustaste einen Rahmen um einen Teilbereich zieht. Wenn man mit der rechten Maustaste auf die Grafik clickt, kann man die aktuelle Grafik in eine Datei speichern. Dies ist für eine spätere Auswertung sehr praktisch. Darüber hinaus bietet das Auswahlménü auch noch einige zusätzliche Optionen.

Kontakt Daten:

Norbert Redeker DG7EAO, Telefon 0201 4099578, info@redeker-web.de