Arduino VNA antenna analyzer

Autor:
Data de publicació: 28-02-2018

21/06/2015f4goh

Par F4GOH : f4goh@orange.fr

Caractéristiques :

HF : 1 MHz à 60MHz
Analyse en Réflexion et transmission
Utilise le logiciel Jvna de DL2SBA
Mode « Standalone »
Afficheur LCD 4×20
Mode Bluetooth avec Blue VNA
Navigation dans le menu par encoder rotatif
Cout 50€ hors module bluetooth

Le VNA est basé sur les circuits suivants :

Arduino NANO
Module ad9851
ad8302

ebay-links : components list

Pcb disponible

Pcb available mail me : f4goh@orange.fr

Pour le PCB OZ6HR voir le lien suivant : OZ6HR edition

l’implantation et les tests sont les mêmes.

Introduction

Ce VNA HF est une adaptation du schéma décrit sur le site russe de RA4NAL.

(http://ra4nal.qrz.ru/vna.shtml) avec des améliorations au niveau périphérique hardware et logiciel.

En effet vous pouvez utiliser ce VNA à partir de 3 interfaces logicielles différentes :

Mode 1 : Connexion au PC

Utilisation du logiciel JVna de DL2SBA (http://vnaj.dl2sba.com)

Mode 2 : Connexion par un module bluetooth

Utilisation du logiciel blue VNA (https://play.google.com/store/apps/details?id=ro.yo3ggx.btvna&hl=fr)

Mode 3 Utilisation en « Standalone »

Les informations des paramètres de l'antenne sont affichées sur un écran LCD 4×20 caractères. La sélection de la bande ou fréquence se fait avec un encodeur rotatif.

Les informations affichées sont sauvegardées dans un fichier texte sur une mSd card.

Description du schéma :

On reconnait aisément le schéma de RA4NAL, l'Arduino UNO a été remplacé par un NANO facilitant l'intégration de la connexion USB.

L'oscillateur DDS est un ad9851, le signal est amplifié par un ERA2SM avant d'attaquer le coupleur tandem à base d'un BN43-202.

La tension directe et réfléchie du coupleur est injectée dans un ad8302 qui nous donnera une tension continue proportionnelle à la magnitude et la phase des deux signaux.

Ces 2 tensions Vmag et Vphs sont converties en valeurs numériques par l'Arduino avant d'être traitées par la fonction calculDut(int adcMag, int adcPhs)

RL=-20log(Rho)
Rho=10^(RL/-20)
Z=(ZL-ZO)/(ZL+Z0) avec Z0=50ohms
Z=a+jb avec
a=Rho*cos(phi)
b=Rho*sin(phi)
ZL=(1+Z)/(1-Z)*Z0
ZL=RS+jXS avec
$RS=abs(1-a^2-b^2)/((1-a)^2-b^2)$
$XS=abs(2b/((1-a)^2-b^2))$
$|Z|=sqrt(RS^2+XS^2)$
SWR=(1+Rho)/(1-Rho)

Le relais Omron permet activer le mode de transmission et ainsi mesurer le gain d'un filtre intercalé entre la sortie DUT

et l’entrée DET.

La liaison SPI est partagée entre le pilotage du DDS et la sauvegarde des données dans la mSd card. Sachant que l’alimentation de la carte mSd est de 3.3V, Il faudra ajouter des diviseurs de tensions en sortie de l’Arduino.

L’afficheur LCD 24×20 caractères est piloté via le bus I2C à condition que celui-ce soit équipé d’une interface PCF8574 (I2C Serial Interface Module For LCD1602 Display).

On veillera bien à ce que le module bluetooth connecté sur J7 soit compatible 5V au niveau des E/S, car ce n’est pas le cas de tous les modules.

Si vous désirez faire une analyse avec Jvna sur PC , il faudra déconnecter le module bluetooth afin de ne pas provoquer de conflit sur la ligne série.

La Led D3 s’allume quand un transfert de données à lieu sur la ligne série.

Ci-dessous le lien avec l’ensemble des fichiers de montage. Le circuit imprimé a été conçu pour faciliter l’assemblage sur PCB fait maison.

Par la suite une autre version de PCB à été réalisée en format plus réduit, vous pouvez me contacter par mail pour avoir un exemplaire

f4goh.bzh at gmail.com

VnArduino PCB files

Montage et tests:

Commencer par souder le support de l’Arduino nano, le support du DDS ad9851, le connecteur de l’afficheur LCD et l’encodeur rotatif.

N’oublier pas les condensateurs de découplage et anti-rebonds.

Programmer le fichier VnArduino.ino dans l’arduino à partir du lien suivant :

https://github.com/f4goh/VnArduino

Si vous en voulez directement programmer les fichiers HEX, ceux-ci sont disponibles ici . Il faudra utiliser Xloader et ne pas oublier de choisir une vitesse de programmation de 57600 bps.

Maintenez l’encodeur rotatif enfoncé pendant la mise sous tension pour accéder au menu.

Dans le menu, choisir JVNA

Une fois validé, l’arduino reset alors tout seul en mode Jvna

Lancer Jvna sur votre ordinateur, choisir le bon port de communication dans le menu « driver sélection » ainsi que « MiniVna ». Faire un test de connexion et cliquer sur update.

Effectuer une calibration en réflexion factice pour le moment, puis dans le menu Fréquency calibration ajuster le DDS ticks 23 860 477 pour être compatible avec l'ad9851.

Aller dans le menu Générator, cliquer sur « on air », puis vérifier la fréquence générée en sortie du DDS avec un oscilloscope.

Souder maintenant l'amplificateur ERA-2SM ou un équivalent de votre stock de composant.

Vérifier à nouveau le signal en sortie de l'amplificateur.

Assembler et souder maintenant le pont de mesure et les atténuateurs en T.

Sur charge 50 ohms, vérifier avec un oscilloscope les tensions aux bornes de R12 (tension directe max) et aux bornes de R6 (tension réfléchie quasi nulle).

Sans charge (sortie ouverte), vérifier avec un oscilloscope les tensions aux bornes de R12 (tension directe max) et aux bornes de R6 (tension réfléchie max).

Ci dessous la tension aux bornes de R12 sous charge 50 ohms

Terminer par souder les composants autour de l'ad8302.

Vérifier avec un voltmètre la tension de référence VREF délivré par ad8302 qui doit être de 1.8V.

Le VNA est prêt à être utilisé.

Refaire une calibration, cela doit correspondre à la copie d'écran ci-dessous.

Pour utiliser le mode « standalone », il sera nécessaire d'effectuer une calibration (menu à la 4eme ligne de l'afficheur LCD).

L'emplacement des résistances calib sont utilisés pour ajuster la mesure sur charge 50ohms pour avoir RS=50 ohms.

Il faut alors placer une seule résistance d'une valeur comprise entre 300ohms et 2Kohms sur l'un des deux emplacements, mais peut-être en n'aurez-vous pas besoin.

Roland (ON7UF) a ajouté une résistance R22 de 820 Ohm qui lui donne une calibration exacte pour 0, 25, 50, 100 et 200 Ohm à 10Mhz. Il précise aussi que la mesure de longueur de câble fonctionne dans la nouvelle version 3.1.8 de vna/J.

Amélioration de de la chaine de mesure par F1AIA :

Ci-dessous les schémas comportant les modifications à appliquer au niveau des valeurs d'atténuation ainsi que sur la partie amplification avec le remplacement du ERA2 par un SGA3386. En appliquant ces corrections , on peut mesurer des atténuations jusqu'à -40dB et des valeurs de S11 ou Return Loss (RL) de -35 à-39dB.
La dynamique de l'ensemble est améliorée grâce à une meilleure amplification du SGA , accompagné par un meilleur choix des valeurs de référence appliquées sur les entrées de l'AD8302.

Merci Jean-Claude pour ces infos.

SGA 3386 avec sa résistance de 68 ohmsLes atténuateurs modifiésl'atténuateur 8 dB sur l'entrée DET

mise à jour hard et soft sur github

Installation du Module Bluetooth

1- Procurez-vous un module HC-06

2- Suivre le tutoriel de Martyn Currey afin de configurer la vitesse en 115200 Bauds (AT+BAUD8) (programme bluetooth_config.ino)

3- Effectuez le câblage suivant :

Il est possible d'associer le couple de résistance (1K,2K) comme le montre le tutoriel de configuration ou (1.8K,3.3K).

4- Charger le programme VnArduino_HC06.ino ou VnArduinoSd_HC060.ino , en fonction de l'utilisation ou non de la carte mSD.

5- Une fois le programme implanté dans l'Arduino Nano, connecter le module bluetooth. celui-ci clignote.

Attention lorsque le module bluetooth est relié à l'Arduino, celui-ci prend la main sur la connexion USB. Si vous voulez éviter de débrancher le module bluetooth lors d'une utilisation avec JVna, insérer in interrupteur sur la ligne TX (fil vert) du module bluetooth.

6- Associer le module Bluetooth dans votre tablette.

7- Lancer le programme Bluevna, choisir le vna STD (miniVna)

8- Sélectionner le module hc-06 lorsque la fenêtre apparait.

9- Ne pas oublier de lancer une calibration. consulter le manuel ici

Si vous avez un doute sur la commande, faites un test avec le menu Générator afin de mesurer sur la sortie DUT un signal de fréquence fixe.

Bonne bricole

73

Anthony

Il est maintenant possible d'utiliser le VNA en balise HF.

Il sera possible de transmettre en WSPR, PSK, RTTY, CW etc… et ce, sur 3 bandes alternativement.

le code source est inspiré de l’article de la balise HF multimode

Ajouter une horloge en temps réel DS3231 sur le BUS I2C (en parallèle avec l’afficheur LCD) (GND, VCC, SDA, SCL)

Charger ensuite l’Arduino NANO avec le programme HF_Beacon_1509_VNA.cpp.HEX à l’aide de XLoader.exe (57600 bps)

Puis, installer le programme de configuration HF Beacon configuration

Lancer le logiciel de configuration en suivant l’exemple ci dessous :

Établir une connexion avec la balise sur le bon port COM,
Changer principalement le CallSign, le locator et le mail.
Choisir le mode de fonctionnement de la balise (Continue, Every, Hour, Sweep)
Sélectionner les bandes et les QRG
La puissance n’est pas gérée pour la carte du VNA
Choisir le mode de transmission
Changer le texte si besoin (attention 250 caractères max)
En WSPR ne pas oublier de faire une mise à jour de l’horloge (SYNC NTP)
Cliquer sur « Send to beacon » afin de charger la configuration dans l’Eeprom de l’arduino Nano.
Pour finir « Run Beacon »

Un exemple de configuration est dans le fichier (HF_Beacon_1509_VNA.ZIP (f4goh_win.bcn)

Vous pouvez calibrer votre DDS en ajustant la fréquence de calibration 180000000 Mhz. (Dans le cas ou la fréquence d’émission est pas tout à fait bonne)

L’émission se fait sur la sortie DUT.

La mesure du SWR est possible. L’option ATU (Auto Tunner) est désactivé.

La puissance étant relativement faible (<100mW), il faudra ajouter un amplificateur en sortie DUT. Vous pourrez faire des essais en décodant les trames avec un récepteur et un PC.

Un PA sera bientôt décrit sur cette page…

Réalisation OM :

Félicitations à Roland (ON7UF) pour son adaptation.

Congratulation Rainer (OE1KFR)

Un nouvel assemblage de Jeff (F5BCB)

Montage de Gérard (F6BSZ)

Montage de Claude F8BOJ

Publicités