Credits © 2014,2017 -Doug WA3DSP

steve@walch.net **BAYCON 2018**

ANNUAL BAY-NET RADIO CONFERENCE

Building a Portable Raspberry Pi2 Asterisk AllStar Node

Steve Walch
W6MNL

Modifying a Baofeng UV82 for use with ALLstar

Introduction

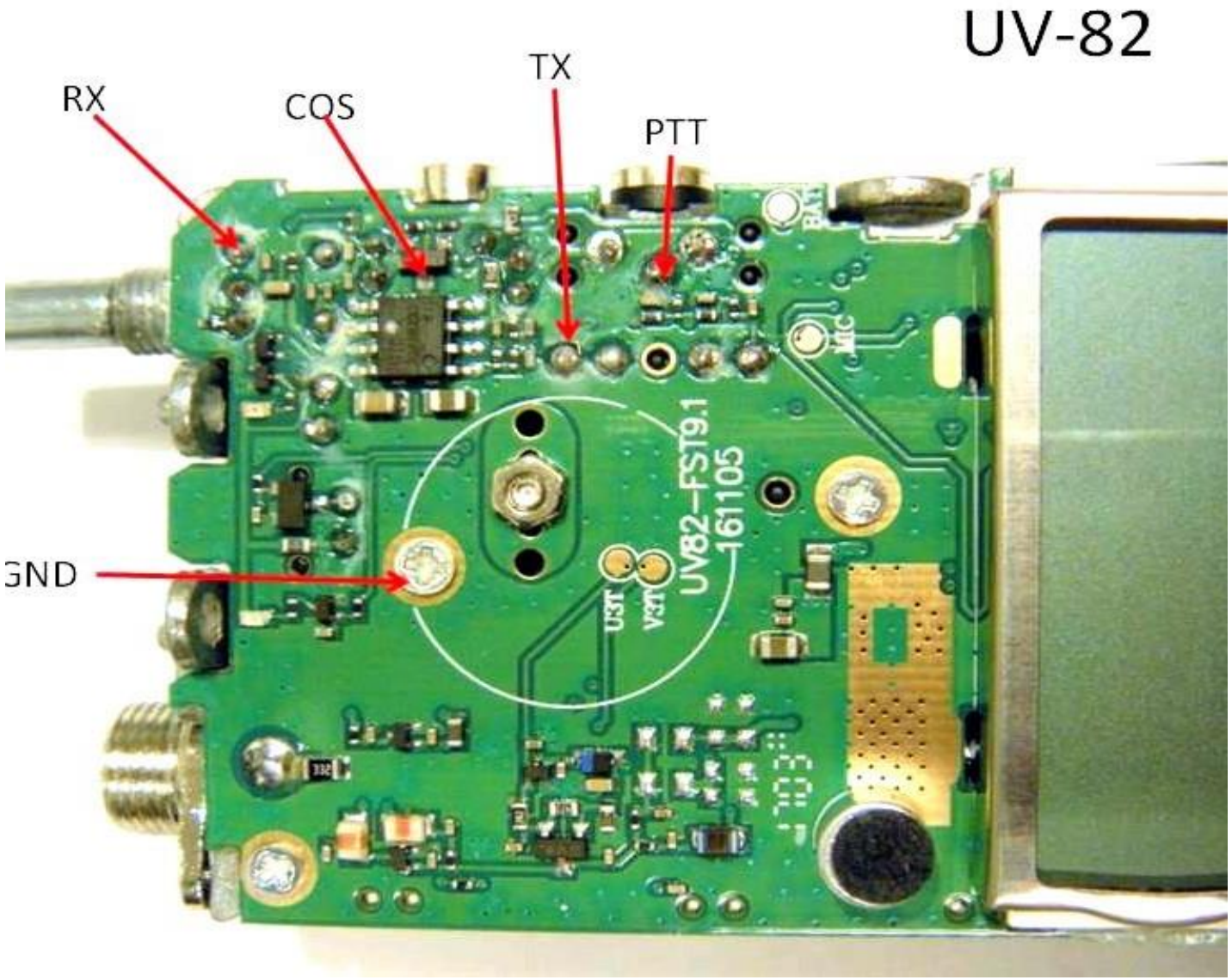
- The Baofeng UV-82 is a dual band 144/440 1W or 5W radio. It is typically available for less than $30 from many retailers. It makes a nice Allstar node radio which is a little easier to modify and better quality than the Baofeng 888 and it gives you the ability to use either the 2 meter or 3/4 meter bands.

Parts required

- There are very few parts required for this modification. Three resistors, a FET, a connector and shell, and wire is all that is required. The board does not need to be removed from the frame and connections are easy to make
- 4.7K resistor (2)
- 10K resistor
- 2N7000 FET
- DB25 Male and shell
- 4 conductor shielded wire, length of your choice

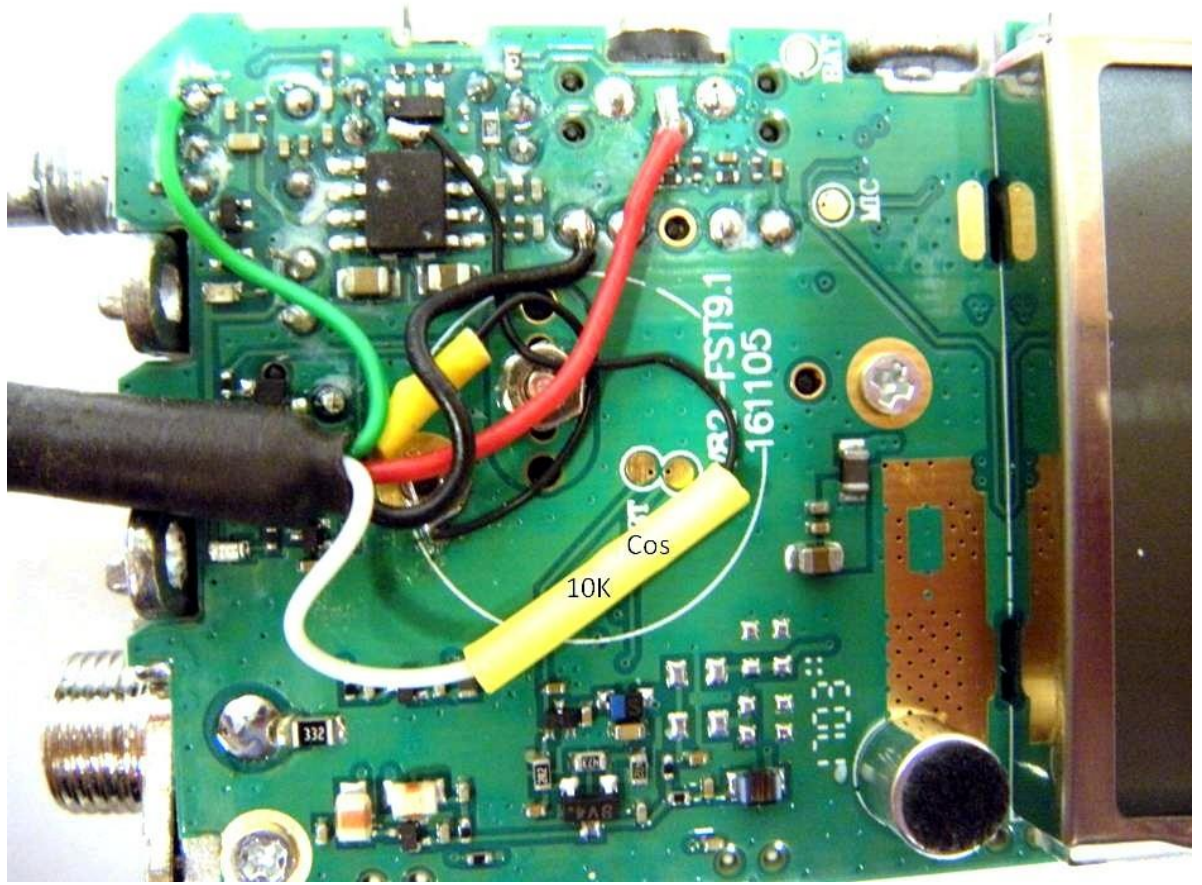
Modification Photos

Here are the connection points on the UV-82 circuit board

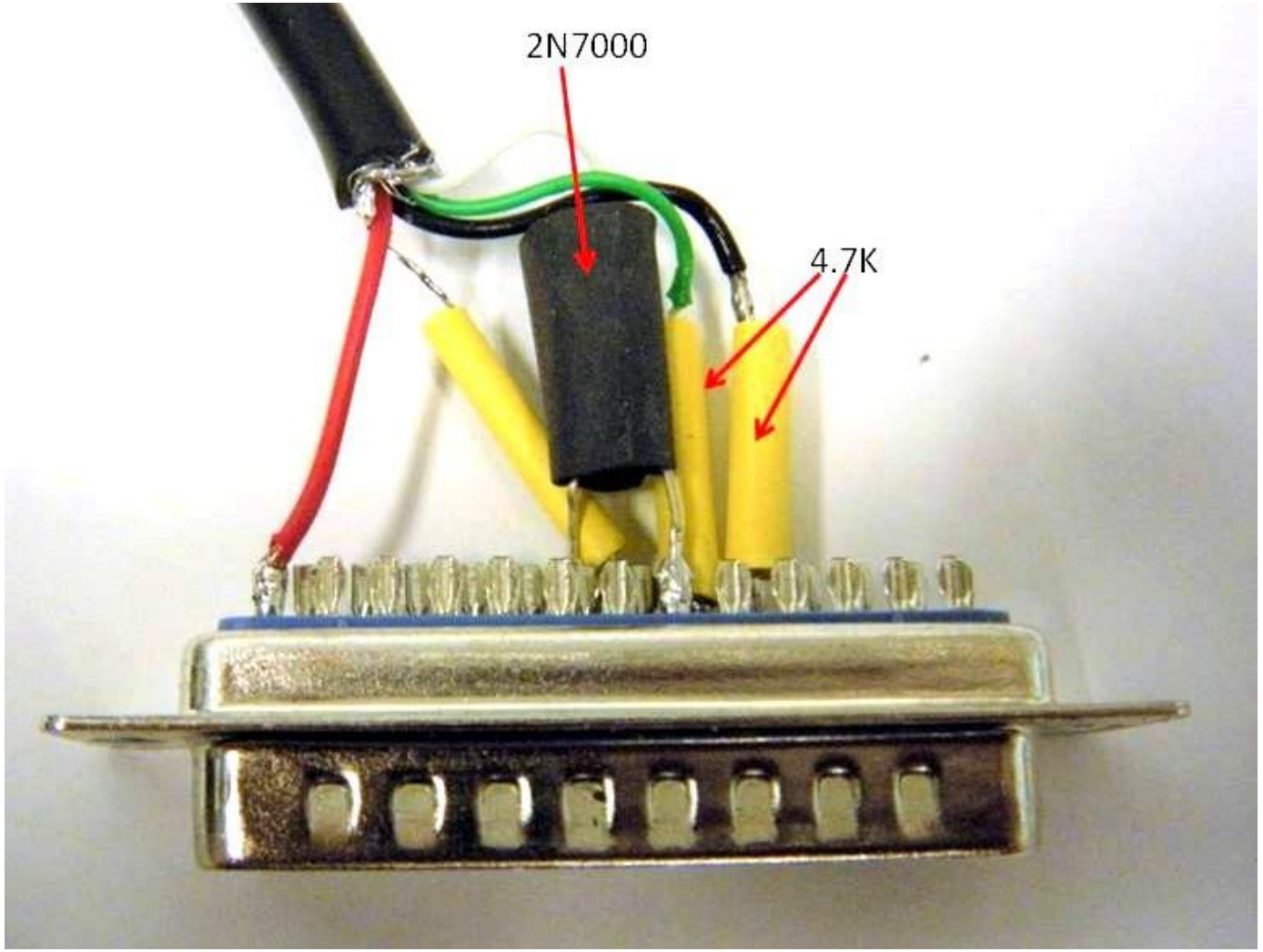


and the wires after connection. Note the COS line is fed through a 10K resistor located in the yellow heatshrink. This can be a 1/4 or preferably a 1/8W resistor. Wire colors are not important as long as you standardize at both ends.

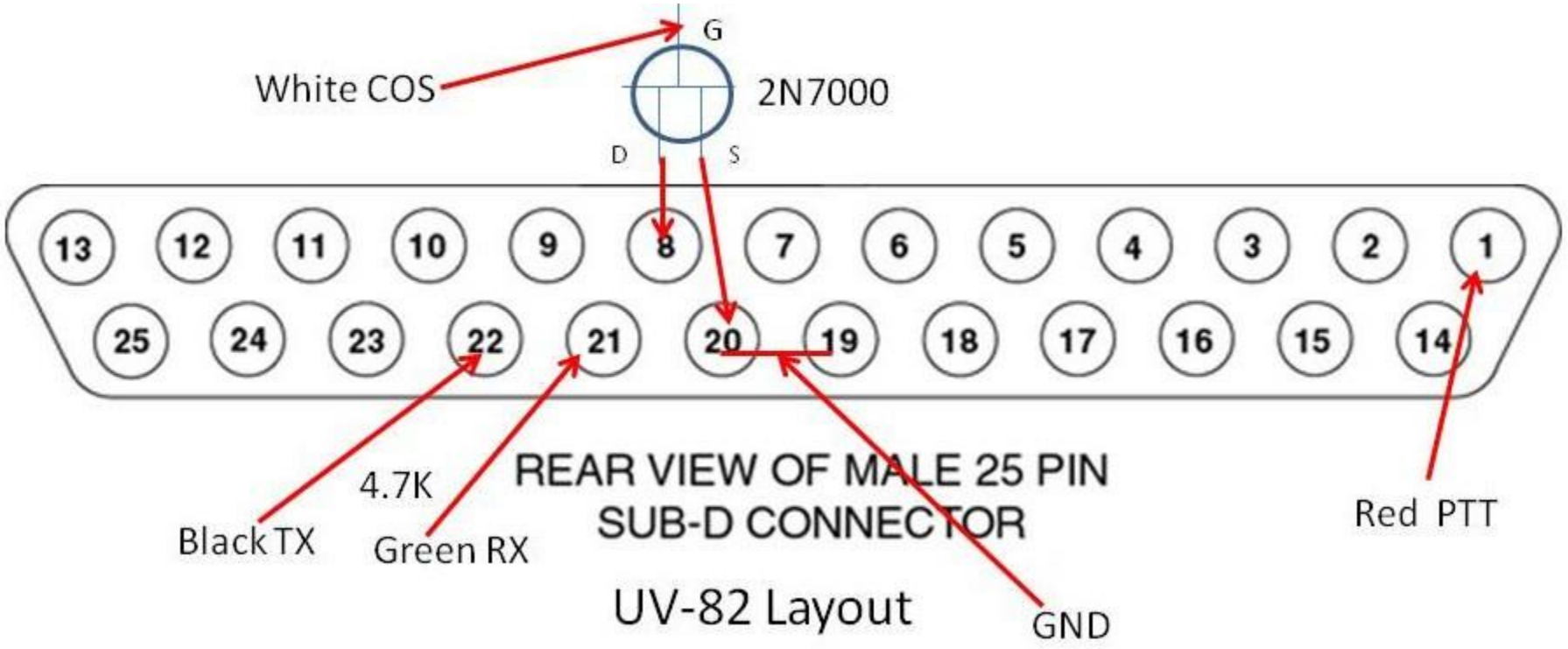
Shown here the Red is PTT, Green is RX audio, Black is TX audio, White is COS with small black on the other side of the resistor to the board, and the small black is ground. is ground.



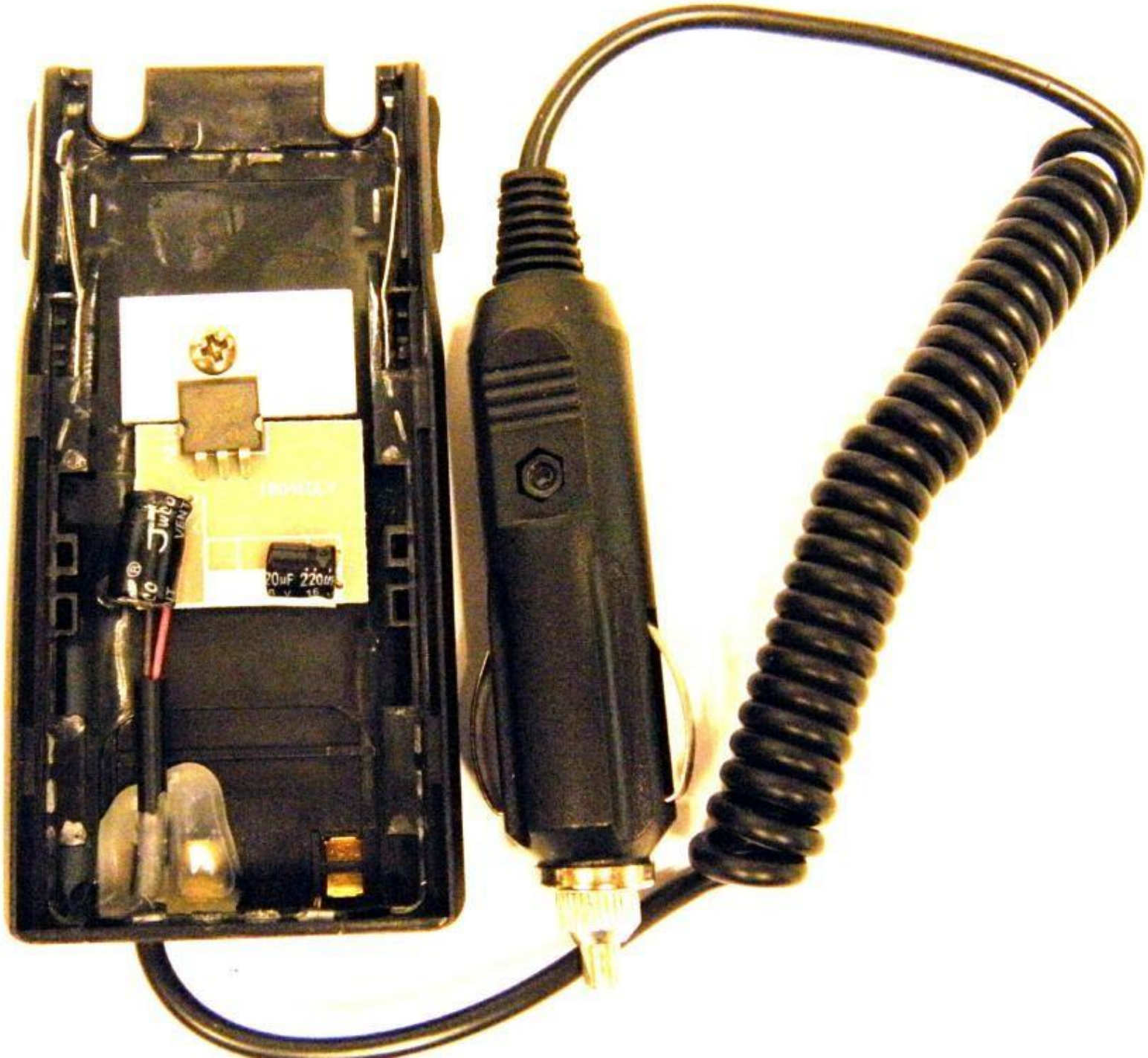
This is a view of the DB25 connector end wired to plug into a DMK/URI or similarly wire mate. The 4.7K resistors and the 2N7000 FET are mounted at the connector end



Here is the pinout diagram of the DB25 connector. The source pin of the 2N7000 goes to ground, the drain goes to the COS pin 8, and the gate goes back to the radio to the 10K resistor.

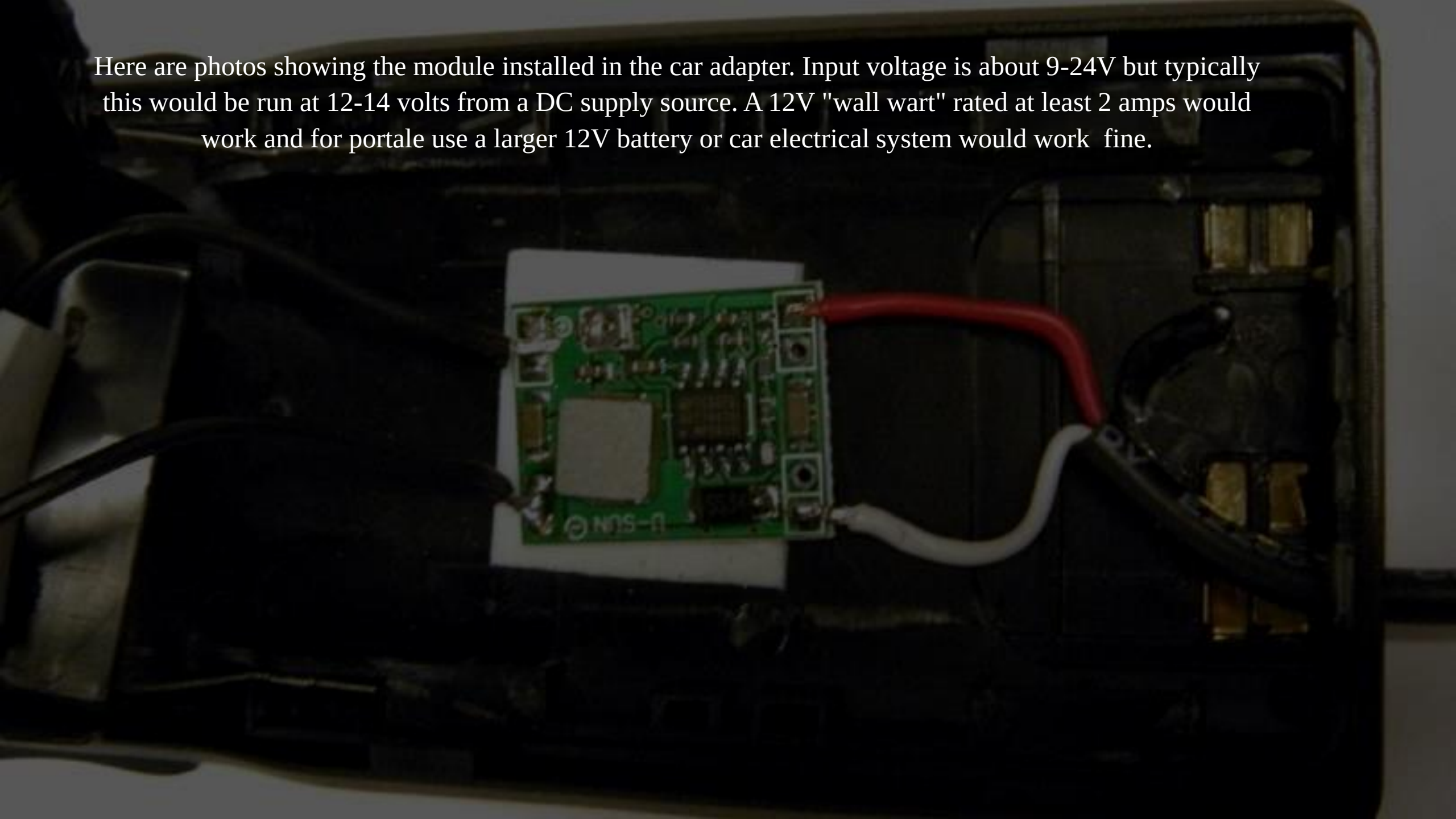


and here is a photo of the completed assembled radio ready to play. It is highly recommended and often necessary that an external antenna (removed from the radio) be used to keep RF away from the radio.



The UV-82 being a handheld is normally powered by a battery but as a node radio you would need a steady fixed source of power. The 12V battery eliminator adapter for the UV-82 is one way to do this but the commercial adapters are not well designed electrically. They use an inefficient three terminal regulator and an inadequate heatsink. This quickly heats up and would destroy itself in short order. The answer is to open the car adapter, remove the original regulator, and install an efficient replacement regulator. Here is what the adapter looks like before removing the original parts.

Here are photos showing the module installed in the car adapter. Input voltage is about 9-24V but typically this would be run at 12-14 volts from a DC supply source. A 12V "wall wart" rated at least 2 amps would work and for portale use a larger 12V battery or car electrical system would work fine.



Programing

- The easiest way to program your UV-82 is using a programming cable and Chirp. Front panel programming of a Baofeng is a chore but can be done. You also need to disable the voice prompts so they do not go out over the node. Once programmed you can change the channel on the radio. The BOTTOM display is what is received and transmitted. In simpleusb you would use rxboost=1, cosfrom (carrierfrom) =usbinvert, and the audio levels would need to be set. A good starting point is RX level = 550, and TX level = 750 but these could vary greatly depending on the FOB type you are using and the radio you are using to talk to your node. The volume on the UV82 has no bearing on Allstar levels. It should normally be turned all the way down but you can raise it to monitor the input signal to your node. A dummy plug should be inserted in the larger remote mic jack on the UV-82 to disable the internal microphone when using the radio as an Allstar node. A standard unconnected stereo miniature plug can be used or preferably a plastic dummy plug cut from a stylus pen like this:

* Plastic dummy plug

*Cut off the pen but leave the string to pull it out when needed.

*CAUTION! Do NOT run this radio on high power on a busy Allstar channel. It will most definitely overheat. Handhelds are not designed for continuous TX operation at high power. It should run fine on low power.

Asterisk Allstar on the BeagleBone Black and The Raspberry Pi 2

Acquiring a RPi 2

- Note that the RPi2 uses a micro USB connector for 5V power. You will need a 5V 1-2A wall wart with a micro USB connector on the cable end or equivalent. This does not come with the board. It is a good idea to also order a case for your board.

- $35 Newark - http://www.newark.com/raspberry-pi/raspberrypi-2-modb-1gb/sbc-raspberry- pi-2-model-b-1gb/dp/38Y6467
- $35 MCM - http://www.mcmelectronics.com/product/83-16530?scode=GS401
- $44 Adafruit - https://www.adafruit.com/products/2358
- $57 AU - "http://www.buyraspberrypi.com.au/shop/raspberry-pi-2-model-b/
- $46.95 Amazon - Free Shipping - http://www.amazon.com/Raspberry-Pi-Model-Project-Board/dp
- /B00T2U7R7I/ref=sr_1_1?ie=UTF8&qid=1425756130&sr=8-1&keywords=rasberry+pi+2
- $35 Allied - http://www.alliedelec.com/raspberry-pi-raspberry-pi-2-model-b/70465426/
- 24.90 UK - http://uk.rs-online.com/web/p/processor-microcontroller-development-kits/832-6274/
- $35 Element 14 - http://www.element14.com/community/community/raspberry-pi
- $34.99 Microcenter ($29.95 in store) - a href="http://www.microcenter.com/product/443781
- /Raspberry_Pi_2_Model_B

RaspberryPi 2

About $33.00

UrI USB Radio Interface

- DMK Engineering is pleased to introduce URI -- the USB Radio Interface designed for linking amateur radio communications through personal computer network channels.
- URI allows a Land/Mobile FM radio device to be connected to a Linux PC via the USB port, extending radio communications across PC network channels. Now users can link two or more radios together with audio and digital control signals that even enable control over remote radios installed in inaccessible locations or remote repeater nodes.
- URI connects to any radio system that brings out audio and transmit control signals.
- About $100.00

Url Picture

Why Allstar?

- We are often asked why Allstar with all of the new RF digital repeaters and their VOIP connection schemes out there.
- The best answer is open source, quality audio, and freedom to do what you want. Dstar, Fusion, DMR, etc are all at least partially if not completely proprietary systems.
- In most cases the VOIP connection system that makes them somewhat like Allstar is proprietary.
- Code plugs, Rooms, etc. all make for a confusing and hard to maintain system.
- It also makes a system which is managed much like IRLP. The individual users has little control other than to connect somewhere.
- Then there is the lousy audio that goes along with the current Amateur Radio digital RF schemes.

More

- The advent of the inexpensive small board computers and very easy scripted setup make getting on Allstar easier than ever.
- A savvy user who has the ability to make some modifications and build some simple circuity could get on Allstar for well under $75.
- A totally plug and play system minus the node radio would be no more than $150.
- For repeater operators Allstar is a complete repeater controller that probably has more flexibility
- Allstar connects RF systems with full duplex quality audio.
- It also gives the user a choice in how and where they connect. Private or public group or individual connections can be setup quite easily.
- If you are in a restricted living environment like a retirement home or HOA that does not allow antennas Allstar gives you the radio feel without an obtrusive antenna or big expense.
- You could setup a simplex node in your apartment or dwelling and have local coverage with in a few blocks to several miles between your handheld and you own personal node.
- Couple that with the ability to setup connections to all your buddies that also have nodes and you have your own talk group with everyone using real radios to access.
- You also have the ability to access via computer, phone, Echolink, and many other methods.

Image File and SD cards

- The image file is sized to go on a 4GB microSD card as a minimum. The compressed image (zip file) is just over 500M. It is strongly suggested to use an 8 or 16G card. I suggest a Sandisk Ultra or Ultra+ class 10 8G or 16G card. The larger the usable area on the card the more space there is for wear levelling. This will reduce the number of writes to any one location on the card and greatly extend its life.
- I have instituted as much write to RAM in this package as possible so there are very few writes to the SD card. After several years of experience with these cards we find them to be very reliable. However they can eventually fail so a backup is always very important.

More

- Backing up an SD card image once you have it configured the way you want it is as easy as reading the image into your computer using Linux 'dd' or Windows 'win32diskimager'. You MUST use an image copy method. You cannot just copy files to make an image backup of the SD card. Keep a second SD card burned with the image near your BBB or RPi2 and in the event of a failure it could be swapped in seconds. This is not something that would happen very often but it is always good to be prepared.
- Starting at BBB version 1.2.1 and RPi2 version 1.0 a complete image backup can be done on the fly to a USB stick, see the backup how-to for details
- We also do not recommend expanding the partition size to the full SD card size unless you have a real reason to do that. It does nothing for reliability and makes it difficult to do image backups as the entire card must be backed up and restored.
- The image uses Archlinux. Archlinux uses systemd for loading modules, maintaining logs, and lots of other things. It is totally different than the initscrpt you have used in many other Linux distros.
- I just want you to be aware of this when you are looking for things that are not there like /etc/init.d! I am including some links that will help you with that but for the most part after you get your image configured it is plug and play. You do not have to be a Linux guru but a little knowledge of Linux and systemd would be helpful in maintaining your BBB or RPi2 Allstar system.

Downloading the image file

- The image file must be downloaded and written to a microSD card. This can be done on a Windows or Linux PC. If you are using Windows and win32diskimager the procedure is to first unzip the downloaded file. You should then have the image (.img) file in your directory. Then insert the card in your reader, start the win32diskimager program, select the file you downloaded and then select write.
- Download win32diskimager - http://sourceforge.net/projects/win32diskimager
- Note that after installing win32diskimager and starting for the first time you may get an error about elevating permission levels or something like that. If you do, exit out and left click once to select the win32diskimager icon, then right click and select "run as administrator" It should then run properly.
- See details on how to use win32diskimager here or at this site
- One of the sites gives an example using the Raspberry Pi. The procedure would be the same for the BBB just substitute the BBB Allstar image. It takes anywhere from 5-15 minutes to write the card depending on the image size and your systems speed. Once the card is successfully written you can remove it from the PC reader/writer and install it in the BBB with the power disconnected. Note that this one image written to your SD card contains the complete Archlinux/Allstar package.
- If you are using Linux to write the image, insert the card in your card reader/writer. Then do a 'df' to determine the device name. On my system it was /dev/sdb but yours may be different. It is important to identify the device properly. You do not want to write to your hard drive!! Once you have identified the device use 'dd' to write the image.

Download of image files for Rpi2/Rpi2-3

- **The latest images are available here -**
- **RPi2-3 Image Version 1.5rc2 - Self Extracting exe for Windows (LATEST) - https://hamvoip.org/RPi2/RPi2-3_V1.5rc2_Allstar.exe**
- **IMPORTANT** - Download this README for the V1.5 version
- **RPi2-3 Image Version 1.5rc2 - xz archive (LATEST) - https://hamvoip.org/RPi2/RPi2-3_V1.5rc2_Allstar.img.xz**
- **IMPORTANT** - Download this README for the V1.5 version
- **RPi2-3 Image V1.0 - V1.02beta (This version has been deprecated - Download one of the latest versions above) - https://hamvoip.org/RPi2/RPi2-3_V1.02beta_Allstar.img.zip**

Booting the image

- Now that you have the microSD card with image inserted in your BBB or RPi2 apply power and it should boot to a login prompt. The next section will give options on how to initially login.

First Communications with the Rpi2

- Starting with BBB version 1.2.1 and RPi2 version 1.0 the IP address is sent out the radio port as voice on initial start. If you have a radio properly connected this might work for you. Not all cases will work as initially the system may not be configured properly for your radio. If not you can use one of the other methods listed below.
- Initially you have to communicate with the BBB or RPi2 to determine its local IP address. One way to do that is to use a terminal connected to the HDMI port and a keyboard connected to the USB port. However there are alternative methods listed below if you don't want to bother with a keyboard and monitor.
- If you do use the keyboard and monitor you do not need a mouse. There is no graphics mode, you login in standard Linux text mode with the keyboard. The RPi2 has a standard HDMI connector. The BBB has a micro HDMI connector so you would need to acquire a micro HDMI to standard HDMI cable or adapter. This does not come with the BBB but could be purchased locally at a Walmart, Best Buy, etc. or on the Internet. Most newer monitors now have HDMI inputs. If not there is an HDMI to DVI adapter available if your monitor has a DVI input. If you do not have an HDMI or DVI input on a computer monitor you can use any digital broadcast television receiver as they all have HDMI inputs. This connection will only be needed for initial setup.

Firewall

- A firewall has been added but NOT enabled by default. In most all cases where you have a router and the BBB or RPi2 is nat'ed behind it a firewall will not be required. If you do initiate the firewall it has the following default rules:

- Ports 4560-4590 UDP - use for iax if needed Ports 5198,5199 UDP - Echolink
- Port 5060 - SIP - Remove comment in file to allow Port 80 - HTTP - Remove comment in file to allow Port 222 - SSH - Allowed
- Port 1194 - openvpn - configure to your requirements All other incoming ports are blocked

Additional Notes

- The openvpn package is now installed for your convenience. Most will probably not use it but it can be a valuable asset in connecting your distant nodes securely. By default it is not enabled. You must setup the configuration files. There are example files in /etc/openvpn and further information on its use can be found at openvpn howto and documentation. See the howto's for information on enabling.

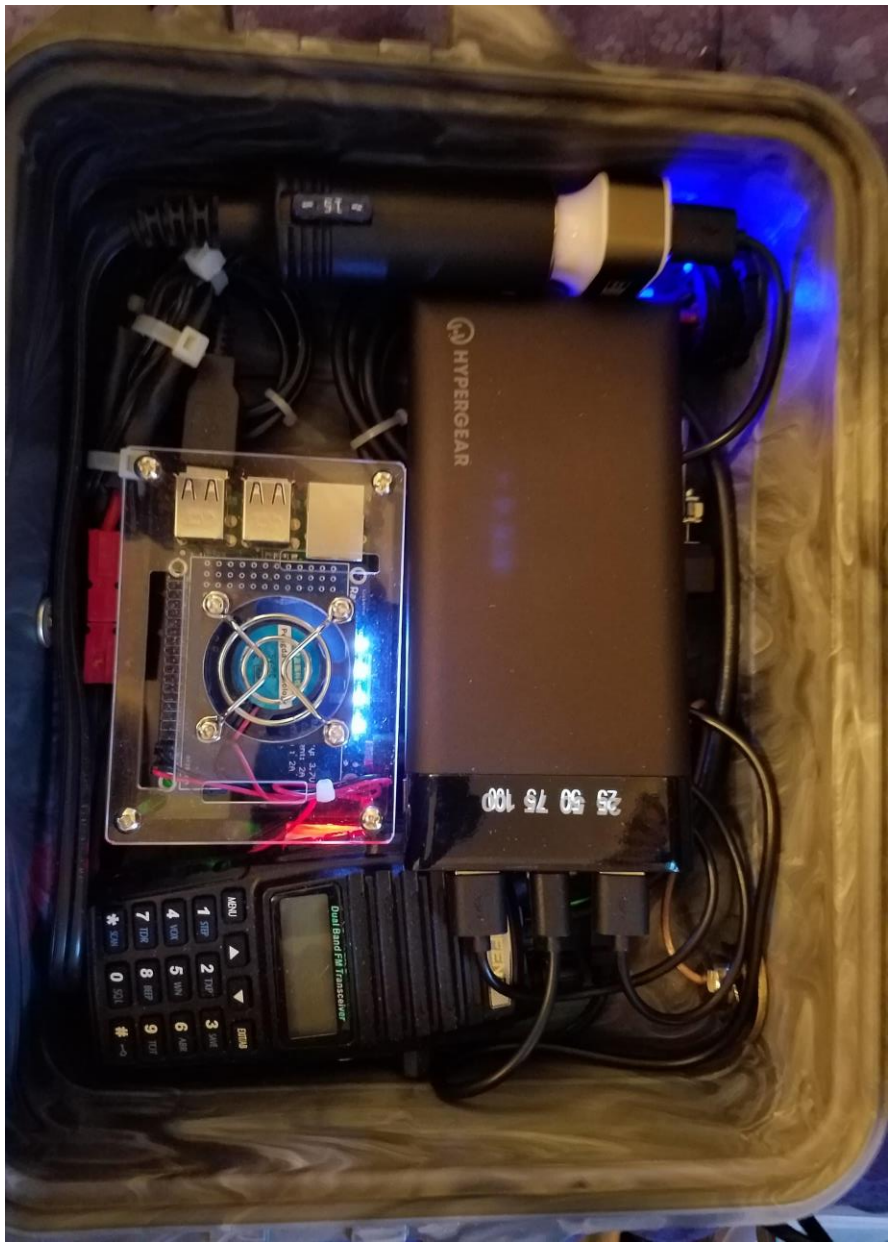
Links

- Archlinux Beginners Guide
- Systemd Users Guide
- BeagleBone Black WIKI
- BeagleBone Black Accessories
- BeagleBone Black Reference Manual
- Linux ate my RAM
- Allsarlink.org main site
- Configuring a two node Allstar system
- simpleusb.conf
- DMK URI USB interface
- USB Allstar Adapters
- Acid System Admin Manual (old but useful)
- Mac SD card utility

Credits

- Mihhail Jakovlev, ES1BIS - Mihhail got me started with all this. Without his sharing of the initial compilation in RPi Pidora I would probably not have gotten started.
- Dave McGough, KB4FXC - Dave and I got together after I moved to the BBB and Archlinux. His input and testing was a great help. Dave fixed the boot problem and was always there when I needed help and he has contributed greatly to the refining of the BBB image. We will continue to work together on future projects.
- Chris Kovacs, W0ANM - Chris is responsible for getting the WIFI operational and documented and for many of the neat scripts like weather, TTS, backup and restore, etc. He has been a great help in testing and project discussions.
- Credit also to Jim Dixon, WB6NIL and Tim Sawyer, WD6AWP for initially contributing scripts for node and net setup.

* Doug Crompton WA3DSP https://hamvoip.org/



Finished Node Pictures Rev1

[functions1998] 1=ilink,1	;	disconnect link = *1<node>
2=ilink,2	;	monitor link = *2<node>
3=ilink,3	;	connect link transceive = *3<node>
4=ilink,4	;	remote command = *4<node>
5=macro,1	;	execute macro = *5<macro#>
70=ilink,5	;	system status
71=ilink,11	;	disconnect permanently connected link =
72=ilink,12	;	*71<node> connect link permanent monitor = *72<node>
73=ilink,13	;	connect link permanent transceive = *73<node>
75=ilink,15	;	play full system status
76=ilink,6	;	disconnect all links
77=ilink,16	;	reconnect previously disconnected links
78=ilink,18	;	permanently monitor link – local only =*77<node>

Questions ?

Thanks Steve Walch

W6MNL

steve@walch.net