KC3ESS Repeater

How we Setup the Maryland DSTAR Image On a Raspberry Pi

A Detailed Setup Guide

This documentation assumes you're setting up a gateway to an existing analog repeater with a DRC board attached and that you're only setting up one repeater module with a DVRPTR_V1 board.

Author: Tyler D Morris N3TDM
Contributions by: Ted Leonard W3VG

Published Fall 2016 at: http://kc3ess.tdmorris.com/

We are not liable and cannot be held responsible for any kind of damage resulting from the use of this setup guide.

Table of Contents

Setup Maryland DSTAR Image

In this guide I will walk you through how we setup the Maryland DSTAR image software and hardware for the KC3ESS Repeater in Bradford, PA.

This guide is based on the Maryland DSTAR Image version: R20151101 4.1.7-V7+

Register with the ircddbGateway

In order to use ircddb you must first register your repeater callsign. Only public hotspots that are allowed to run unattended 24/7 are allowed to register. Approval may take a day or two.

Go to this website to register: http://regsrv.ircddb.net/index.htm

Download the Image

Download the image here toward the bottom of the page:
http://maryland-dstar.org/html/raspiberry.html

Make sure you have a fast Internet connection, as the file is around 3GB in size.

After downloading the image, unzip it or extract the files either using a program like WinZip or use Windows's extraction tool by double clicking the *.zip file you downloaded and in the top of the File Explorer window select "extract all files". This will extract the files from the ZIP file to a folder either of your choosing or the default, which is a folder with the same name as the ZIP file in the same folder you originally downloaded to.

Image the SD Card

We used a 16GB SanDisk Ultra MicroSDHC UHS1 Class 10 card for our Raspberry Pi 2. This can currently be found at many stores including Amazon:
https://www.amazon.com/dp/B010Q57SEE/

The Maryland DSTAR guide tells us to ONLY use SDImager to restore the downloaded image to an SD card. The program can be found here:
https://sourceforge.net/projects/sdimager/

After downloading SDImager, unzip or extract the program by using a program like WinZip or use Windows's extraction tool by double clicking the *.zip file you downloaded and in the top of the File Explorer window select "extract all files". This will extract the files from the ZIP file to a folder either of your choosing or the default, which is a folder

with the same name as the ZIP file in your Downloads Folder or wherever you downloaded the ZIP file to.

1. Open the program by double clicking it in the extracted folder. You may be prompted by Windows User Account Control (UAC) to allow the program to make changes to your computer. Click "yes" or "allow."
2. Make sure your SD card is in your computer's SD card slot. Since we're using the MicroSD card, we'll need to put the tiny MicroSD card in a standard size SD card adapter.
3. Verify in "My Computer" or "This PC" the drive letter of your SD Card. **THIS IS EXTREMELY IMPORTANT**
4. In SDImager, select your SD card drive, the letter from step 3. Make sure this is the correct drive letter, as the program will completely erase the drive.
5. In SDImager select the extracted Maryland DSTAR image file by clicking the button with 3 dots in the section labeled "Image File."
6. Again verify everything is correct and click the "Write" button.
7. This will probably take about 10 to 15 minutes to complete.
8. After the imaging is completed click the okay button.
9. Next go to "*My Computer*" or "*This PC*" and right click the SD card and click "Eject." You should receive a notification saying it is safe to remove the card.
10. Remove the card and insert the MicroSD card into your Raspberry Pi.

Setup the Basics

1. According to the Maryland DSTAR Setup Guide, if you're using a DVAP, DVRPTR_V1, or GMSK board, plug the USB cable into one of the LOWER USB ports on the Pi.
2. Plug in all of the other cables (mouse, keyboard, monitor, Ethernet) except the power cord; make sure your monitor is turned on.
3. Now plug in the Pi's power cable.
4. The Raspberry Pi should boot up and eventually you'll see the desktop.

As you can see above, we chose to use a physical monitor for the setup portion instead of using VNC Viewer. If you do not have a monitor to connect your Pi to, you will need to follow the instructions in the Maryland DSTAR Setup Guide to use VNC Viewer. In the end we have used VNC viewer to make adjustments. Using VNC Viewer, connect to the IP address of your Pi and the password is "*maryland-dstar*" without quotes.

Setup the Internet Connection

We need to setup a static IP address. Here is how to do that.

1. On the desktop of the Pi all items are single left click to open them.
2. On the desktop of the Pi, click "*LXTerminal*" to open it.
3. Next type without quotes, "ip addr show". This will give you the current IP address of the pi. It will look like the following: XXX.XXX.XXX.XXX. It will NOT end in a 0, 1, or 255.
4. Make note of the current IP address.
5. In the bottom right of the desktop, there will be an icon for the network that looks like a computer. Click it.
6. Next click the "*Properties*" button.
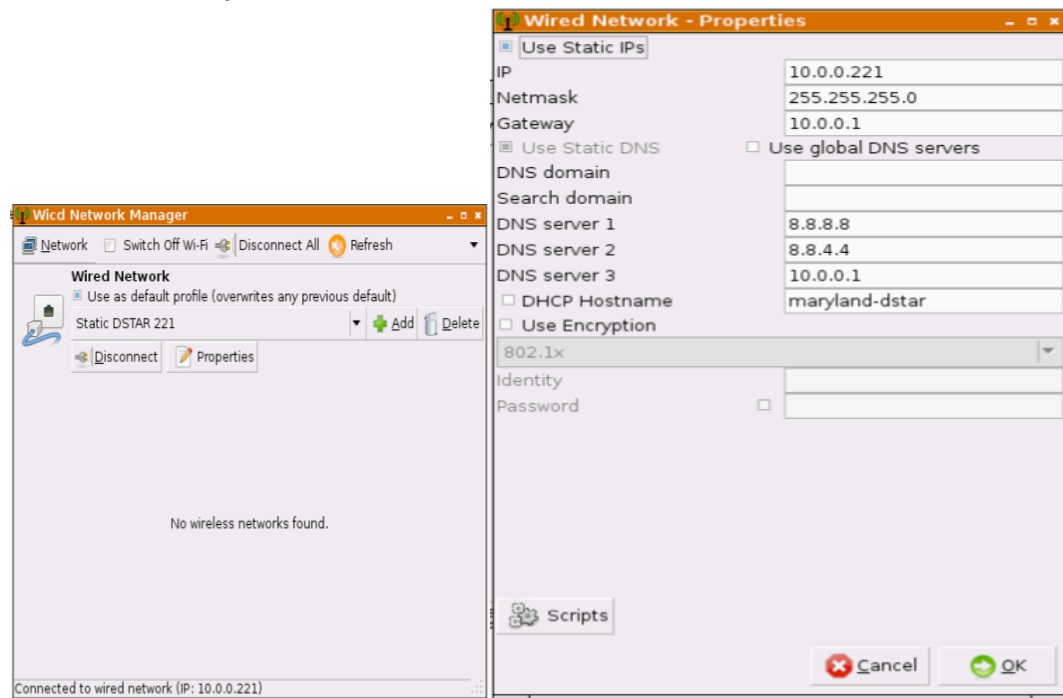
7. In the "*Properties*" window that opens, you'll assign the static IP address.
8. Check the "*Use Static IPs*" check box.
9. In the "*IP*" textbox type the IP address you wish to. Get the first three octets from step 3 and the fourth octet will be whatever you choose that's not assigned to another device on your network.
10. In the "*Netmask*" textbox, enter your netmask, on a home network this will be "*255.255.255.0*".
11. In the "*Gateway*" textbox, type the IP address of your router, use the first three octets from step 3 and in the fourth octet type the number "*1*".
12. We used Google's public DNS servers. In the "*DNS Server 1*" textbox type "*8.8.8.8*".

13. In the "*DNS Server 2*" textbox type "*8.8.4.4*".
14. In the "*DNS Server 3*" textbox type your router's IP address.
15. Then click "*OK*".
16. Make note of the router's IP address as we'll use it in the port forwarding step.
17. Make note of the static IP address you assigned to the Raspberry Pi, as we'll use it in the port forwarding step.

Dashboard Port Number

1. On your Pi's Desktop, click "*LXTerminal*" to open it.
2. In LXTerminal type without quotes, "sudo *etc/init.d/lighttpd stop*" and press enter. This will stop the server.
3. Next, type without quotes, "*sudo nano /etc/lighttpd/lighttpd.conf*" and press enter.
4. This will open a file with text in it. You need to look for the line that says "*server.port = 80*" and change the number "*80*" in that line to "*8082*" without quotes.
5. Next press the keyboard combination "*CTRL+X*" to exit.
6. You'll be prompted to save the file. Press "*Y*" to save it.
7. Lastly restart the server by typing without quotes, "*sudo /etc/init.d/lighttpd start*" and press enter.

Setup Port Forwarding on Your Router

Port forwarding is used so that when a station or repeater tries to connect to your repeater through the Internet, the router knows which IP address to send that connection to. Do these steps from another computer.

1. Go to your router's web configuration page by typing the router's IP address in your address bar of your preferred web browser prefixed by *"http://"*. (Example: http://192.168.1.1) and press enter or go.
2. Login to your router using your username and password. If you did not setup your router or don't know the password, you can check the bottom of the router for a sticker that may have that information on it. Or you can try the default usernames and passwords found at the following website: **http://portforward.com/default_username_password/**
3. Scroll down the page, select your router's manufacturer from the list and try the given default usernames and passwords.
4. After you login to your router, look for a tab that says something like "*Applications & Gaming*," "*Advanced*," "*Security*," "*Port Forwarding*," etc. Click this tab.

5. Next you'll need to forward the following ports and port ranges to the static IP address you setup for the Pi. Chart courtesy of the G4KLX Gateway Manual.

Name	Type	Port Numbers	Description	Comments
DExtra	UDP	3001	DExtra gateway and reflector linking. Also used by DEXTRA_LINK if enabled.	
DCS	UDP	30051 - 30059	DCS gateway and reflector linking, also used by DCS_LINK if enabled.	
DPlus	UDP	20001 - 20009	D-Plus gateway and reflector linking	
CCS	UDP	30061 - 30065	CCS callsign or repeater routing	
G2 Routing	UDP	40000	For callsign and repeater routing	
SSH	TCP	222	Secure Shell	Only if you need remote terminal access outside your home.
Dashboard	TCP	8082	Repeater Dashboard	Port can be changed
Remote Control	UDP	Will be set in the IRCDDB Remote Tab.	For the remote control protocol	To use the IRCDDB Remote Android App.

Configure the G4KLX Software

The following steps show you how to configure and setup the G4KLX Repeater and IRCDDB Gateway software.

Configure DSTAR Repeater Program

First we need to enable repeater one.

1. On your Pi desktop, click "*LXTerminal.*"
2. In the terminal window that opens, type the following without quotes, "*RepeaterOne enable*" and press enter.
3. You can close LXTerminal for now.

Next we need to configure repeater one.

1. On your Pi desktop, click "*Config Repeater One rstr daemon.*"
2. In the window that opens there will be multiple tabs to configure. The instructions that follow pertain to configuring Repeater One.

Repeater Callsign Tab Setup

1. In the "*callsign*" box type your repeater callsign.
2. In the dropdown box next to the "*callsign*" box select the letter that corresponds to the module you're setting up ("B" for 70cm or "C" for 2m).
3. In the "*Gateway*" box, again, type your repeater callsign.
4. In the "*Mode*" dropdown box, select "*gateway.*"
5. In the "*Ack*" dropdown box, select "*off.*"
6. In the "*Restrict*" dropdown box, select "*off.*"
7. In the "*RPT1 Validation*" dropdown box, select "*on.*"
8. In the "*DTMF Blanking*" dropdown box, select "*on.*"
9. In the "*Error Reply*" dropdown box, select "*on.*"
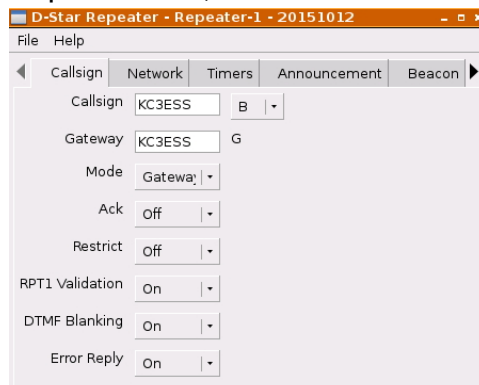
Repeater Beacon Tab Setup

Click the "*Beacon*" tab to setup the Beacon.

1. Use the slider to set the interval between transmissions of the ID beacon.
2. Enter your repeater callsign in the "*Message*" box with a space and the letter of the module ("*B*" for 70cm or "*C*" for 2m).
3. In the "*Voice*" dropdown box, select "*enabled.*"
4. In the "*Language*" dropdown box, select your preferred language.

Repeater Modem Tab Setup

Click the right arrow in the top right next to the "*Beacon*" tab to move to the "*Modem*" tab.

1. In the "*Modem*" dropdown box, select the modem you're using. In our case it's the "*DV-RPTR V1.*"
2. Click "*Configure.*"
3. Now we need to know the port for the DVRPTR board.
   a. Leave the Modem tab open and go to the Pi desktop and click on "*LXTerminal.*"
   b. In LXTerminal type "*sudo dmesg –C*" and press enter. This clears the dmesg log.
   c. Next unplug your DVRPTR board's USB cable and plug it back in.
   d. Then in LXTerminal type, "*dmesg.*"
   e. Look for the port that is associated with the board. It should look similar to "*/dev/ttyACM0*". It may even be the same.
4. In the "*Port*" dropdown box, select the port that you found in the previous steps.
5. Unless you know whether or not you need TX or RX inversion turned on right now, you can leave those settings the way they are.
6. Move the "*TX Level (%)*" slider to something like 20%, then click "*OK.*"
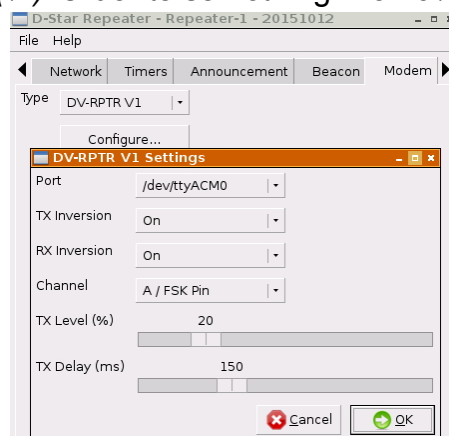
Setup Control Commands

Click the right arrow in the upper right next to the "*Modem*" tab to move to the "*Control 1*" tab.

Repeater Control 1 Tab Setup

In order to setup commands to control the Raspberry Pi from a radio, you need to setup the commands in the RepeaterOne Configuration program on tabs "*Control 1*" and "*Control 2.*"

1. First, in Control 1 tab, you want to select "*enabled*" from the drop down list next to the word "*control.*"
2. Then in the "*RPT1 Callsign*" box type in all caps without quotes "*RPTRCTLX*" where "*X*" is the module letter (typically "B" for 70cm and "C" for 2m). RPT1 Callsign can contain whatever you want, but the letter identifying your module should be in the eighth space. We suggest changing the RPT1 Callsign to something other than "*RPTRCTLX*" (where "*X*" is your module letter). If you use the original and the same commands as outlined in these instructions and the Maryland DSTAR Setup Guide, anyone with RF access can control and potentially shutdown your repeater. For example the "*HALT*" command would require you to physically unplug and plug back in the Raspberry Pi.
3. In the box below it next to "*RPT2*" you need to type your callsign or your repeater callsign depending on what you're setting up, with the letter "*G*" in the EIGHTH space. It MUST be in the eighth space. If your callsign is 5 letters long you need 2 spaces after it then the letter "*G*".
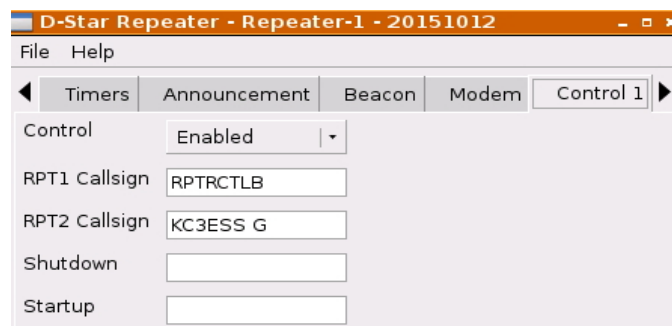
Repeater Control 2 Tab Setup

Click the right arrow in the upper right next to the "*Control 1*" tab to move to the "*Control 2*" tab.

In the "*Control 2*" tab, it lists the commands you can send from your radio in the left text boxes. In the text boxes on the right it lists the shell scripts being called for each command. These shell scripts on the Raspberry Pi are located in "*/usr/local/bin*". If you can write a script in python, you can make the Pi do just about anything based on a command sent to it over the DSTAR radio. You'll find in the "*Control 2*" tab that the basic commands are there, RESTART (restarts ONLY the repeater software & gateway software), HALT (Shuts down the pi - be careful, you must unplug and plug back in the Pi for it to boot again, REBOOT (completely reboots the operating system), SWITCH (if you have a DVLAP or DVMEGA, this can switch the frequency of the board). The documentation also mentions that you can setup a command to send the IP address of the Pi to a cellphone or email address, but it doesn't tell you how. I'm going to tell you how to do it.

1. In the "*Control 2*" tab, replace the "*SWITCH*" command in the left textbox, provided it isn't needed for your particular board, with "*SENDIP*" (no quotes).
2. In the text box to the right, type in the name of the shell script you're going to create. We suggest keeping with the format of the other shell script names. So to the right of the textbox with "*SENDIP*" type in "*klxsendip*" without the quotes. This is what you're going to name the shell script you create.
3. Click File >> Save.

4. Then press "*OK*."
5. Now click File >> Exit.

SENDIP Setup

1. Open a text editor such as LeafPad from your applications menu.
2. Click File, then click Save As, then save it as the name you entered in the previous step (klxsendip) without an extension (so no ".txt" or ".sh").
3. You will need to save it in the Pi folder.
4. You'll need to copy and paste the following code into your file.

   ```
   #!/bin/bash
   sudo python ~/sendipaddress.py
   ```

5. Now save your file by clicking "File" and then click "Save".

Next you need to move that saved file from your "Pi" folder to your "*/usr/local/bin*" folder and set the permissions.

1. Open LXterminal from the applications menu or start menu.
2. Next you need to type in the following command without quotes, "*cd ~/*" and press enter. This ensures you're in the Pi folder where you saved your shell script.
3. Now type the following command without quotes, "*sudo mv klxsendip /usr/local/bin*" and press enter. This will move your shell script into the "*/usr/local/bin*" folder.
4. Next you need to change directories by typing, without quotes, "*cd /usr/local/bin*" and press enter.
5. Then you need to set the permissions on the file by typing the following without quotes, "*sudo chmod ug=rwx klxsendip*" and press enter. This will set the user and group permissions on klxsendip to read, write, and execute. Next we need to set permissions on the same file for others to execute.
6. Type the following in LXterminal without quotes, "*sudo chmod o+x klxsendip*" and press enter.

SENDIP Script Setup

Now that the permissions are set, you need to setup the "*sendipaddress.py*" file. This file is located in the "*Pi*" folder.

1. Go back to LeafPad and click "File", then click "Open", find the file and open it. In this file you'll see a lot of seemingly random text, all formatted into sections.

Under the section towards the top labeled "*Start of Custom Items*", you need to change the "*username*", "*password*", and "*sendtoaddr*" to fit your specific case.

2. In the "*username*" value make sure it says WITH the single quotes, 'XXXX@gmail.com" where XXXX is your username for Gmail." It MUST be a GMAIL account. You may want to create a new email address with GMAIL, just for this purpose due to your password being saved in plain text in the next line.

3. In the "*password*" value make sure it says WITH single quotes, 'xxxx' where xxxx is your Gmail password.

4. In the "*sendtoaddr*" value you can enter your email address WITH single quotes such as 'xxxx@gmail.com' again where xxxx is your Gmail username. This address does NOT have to be Gmail. Feel free to use Yahoo or any other email address you monitor. To send to a cell phone, simply enter your 10-digit phone number with the gateway domain for your carrier. If you have Verizon Wireless you might enter '5555555555@vtext.com' WITH single quotes. If you have AT&T you might enter '5555555555@txt.att.net' WITH single quotes. Here is a list of gateways for various carriers courtesy of http://lifehacker.com/.

> *Alltel: phonenumber@message.alltel.com*
> *AT&T: phonenumber@txt.att.net*
> *T-Mobile: phonenumber@tmomail.net*
> *Virgin Mobile: phonenumber@vmobl.com*
> *Sprint: phonenumber@messaging.sprintpcs.com*
> *Verizon: phonenumber@vtext.com*
> *Nextel: phonenumber@messaging.nextel.com*
> *US Cellular: phonenumber@mms.uscc.net*

The ability to send IP addresses is now setup, however it will only send the IP address that the Pi has on either its Ethernet or WiFi port.

Next I'll tell you how to make the changes to send the public IP address of the Pi.

Changes Necessary to Send Public IP Address

To send the public IP address (Assuming your Pi is behind a NAT firewalled router and NOT in the DMZ. - Most Setups!) you need to make some edits to the "*sendipaddress.py*" file.

1. Open "sendipaddress.py" in LeafPad.
2. Change the following part of the document:

```
fromaddr = username
toaddr = sendtoaddr
rpiname = (socket.gethostname())
message = rpiname +' RPi\'s address: ' #message that is sent
time.sleep(10)

#extract the ip address (or addresses) from ifconfig
found_ips = []
ips = re.findall( r'[0-9]+(?:\.[0-9]+){3}',
commands.getoutput("/sbin/ifconfig"))
for ip in ips:
        if ip.startswith("255") or ip.startswith("127") or ip.endswith("255"):
                continue
                        found_ips.append(ip)
message += ", ".join(found_ips)
```

3. To the following text:

```
fromaddr = username
toaddr = sendtoaddr
rpiname = (socket.gethostname())
message = rpiname +' RPi\'s address: ' #message that is sent
time.sleep(10)  #Time Delay before executing the rest of the script.

#extract the ip address from http://www.echoip.net/
ip = re.findall( r'[0-9]+(?:\.[0-9]+){3}', commands.getoutput("curl
http://www.echoip.net/"))
message += str(ip)
```

Program Your Radio with the Commands

1. In the "*YourCall*" field of your radio type the name of the command in all caps that you put in the textbox to the left of "*klxsendip*" in the "*Control 2*" tab.
2. Next in the "*RPT1*" field of your radio type the "*RPT1 Callsign*" you entered in the "*Control 1*" tab. It's probably "*RPTRCTLX*" where "*X*" is the letter of your module.
3. Finally, in the "*RPT2*" callsign field on your radio type the callsign you entered in the "*RPT2 Callsign*" field in the "*Control 1*" tab. In our case, it's "*KC3ESS G*". Don't forget the "*G*" MUST be in the eighth space.
4. If you want to add the other commands to your radio's memories. Follow the steps above for each command in the list (HALT, REBOOT, RESTART, SENDIP, UPDATE).

Some Notes on SENDIP:

1. Setting up the "*sendipaddress.py*" file will allow it to work (obviously). The important note is that the Pi will send you the IP address EVERY time it reboots the operating system (OS) and EVERY time you restart the Repeater/ircDDB Gateway software.
2. The "*sendipaddress.py*" file is located in the "*Pi*" or "*home*" folder. Get to it in LXterminal by typing without quotes, "*cd ~/*" and press enter.
3. The "*klxsendip*" and other shell script files are located in the "*/usr/local/bin*" folder. Get to it in LXterminal by typing without quotes, "cd /usr/local/bin" and press enter.

Configure IRCDDB Gateway Program

Next we need to configure the IRCDDB Gateway Program.

1. First on the Pi Desktop click "*LXTerminal.*"
2. Next type without quotes, "*Enable gateway*"

ircddbGateway Gateway Tab Setup

1. In the "*Type*" dropdown box, select Repeater.
2. In the "*Callsign*" textbox type your repeater callsign.
3. In the "*Latitude*" textbox type your latitude coordinate with 6 decimal places.
4. In the *"Longitude"* textbox type your longitude coordinate with 6 decimal places. Don't forget the negative if you need it.

5. In the *"QTH"* textbox type your city, state. (Example: Bradford, PA)
6. In the textbox under the "*QTH*" textbox type your country. (Example: USA)
7. In the "*URL*" textbox type your website address or dashboard URL (Example: http://kc3ess.tdmorris.com/)

> Find your latitude and longitude using the following site:
> http://itouchmap.com/latlong.html

ircddbGateway Repeater 1 Tab Setup

1. In the "*Band*" dropdown box, select the letter that corresponds to the band you're using.
2. In the "*Type*" dropdown box, select "*Homebrew.*"
3. If you wish to connect to a reflector on startup, select the reflector in the "*Reflector*" dropdown boxes. Then in the "*Startup*" dropdown box select "*Yes.*" In the "*Reconnect*" dropdown box, select the time in minutes that the repeater will reconnect to the reflector if it is disconnected or linked to another reflector. If this is set to "*Never,*" you will need to manually reconnect if the link is dropped.

ircddbGateway Repeater 1 Tab Setup Continued

Click the second "*Repeater 1*" tab.

1. In the "*Frequency (MHz)*" textbox, type your repeater frequency in megahertz with 5 decimal places.
2. In the "*Offset (MHz)*" textbox, type your repeater's offset in megahertz with 4 decimal places.
3. In the "*Range (kms)*" textbox, type the estimated or known range of the repeater in kilometers.
4. In the "*Latitude*" and "*Longitude*" textboxes type the coordinates of the repeater the same as in the Gateway tab.
5. In the "*AGL (m)*" textbox, type the antenna height above ground level in meters.
6. In the "*QTH*" textbox, type the city, state the same as in the Gateway tab.
7. In the textbox below the "*QTH*" textbox, type the country the same as in the Gateway tab.
8. In the "*URL*" textbox, type the website address or dashboard URL the same as in the Gateway tab.

ircddbGateway ircddb Tab Setup

Click the right arrow next to the "*Repeater 1*" tab until you get to the "*ircddb*" tab.
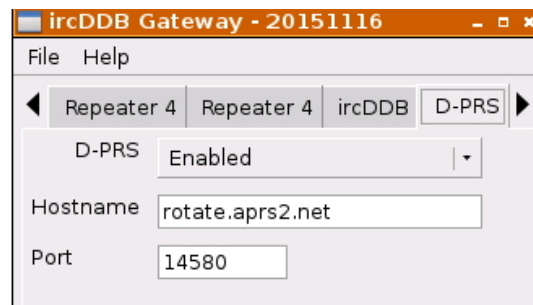
1. In the "*ircDDB*" dropdown box, select "*enabled.*"
2. In the "*Hostname*" dropdown box, select the hostname/sever given to you in the email you received after registering with the ircddbGateway.
3. In the "*Username*" textbox, type the username (usually your repeater callsign) given to you when you registered for the ircddbGateway.
4. In the "*Password*" textbox, type the password sent to you by email after you registered with the ircddbGateway. It will be a long string of letters and numbers.

ircddbGateway D-PRS Tab Setup

Click the right arrow next to the "*ircDDB*" tab to get to the "*D-PRS*" tab.

1. In the "*D-PRS*" dropdown box, select enabled. This will enable your repeater to send its location to the APRS network. It will also be visible at http://aprs.fi/
2. In the "*Hostname*" textbox, use the server for your region. North America uses "*rotate.aprs2.net*".
3. In the "*Port*" textbox, type "*14580*" without quotes.

ircddbGateway DExtra Tab Setup

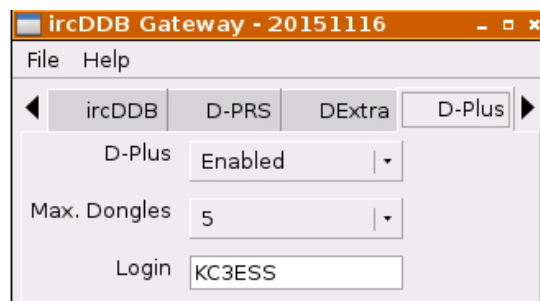Click the right arrow next to the "*D-PRS*" tab to get to the "*DExtra*" tab.

1. In the "*DExtra*" dropdown box, select "*Enabled.*"
2. In the "*Max Dongles*" dropdown box, select "*5.*"

ircddbGateway D-Plus Tab Setup

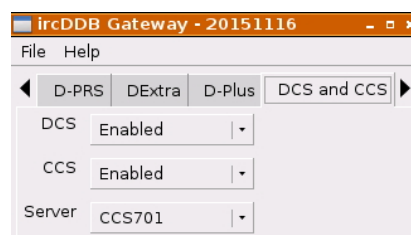Click the right arrow next to the "*DExtra*" tab to get to the "*D-Plus*" tab.

1. In the "*D-Plus*" dropdown box, select "*Enabled.*"
2. In the "*Max Dongles*" dropdown box, select "*5.*"
3. In the "*Login*" textbox, type your repeater callsign.

ircddbGateway DCS & CCS Tab Setup

Click the right arrow next to the "*D-Plus*" tab to get to the "*DCS and CCS*" tab.

1. In the "*DCS*" dropdown box, select "*enabled.*"
2. In the "*CCS*" dropdown box, select "*enabled.*"
3. In the "*Server*" dropdown box, select "*CCS701.*"

ircddbGateway Remote Tab Setup

Click the right arrow next to the "*DCS and CCS*" tab to get to the "*Remote*" tab.

1. In the "*Remote*" dropdown box, select "*enabled.*"
2. In the "*Password*" textbox, type a password that you will remember but is hard to guess.
3. In the "*Port*" textbox, type a port number. We used a port in the 10000 range. This port will need to be forwarded on your router See the section labeled "*Setup Port Forwarding on your Router.*"
4. Next click File >> Save.
5. Then click File >> Exit.

Schedule Automatic Links

Setup Timer Control Preferences

1. On the Pi desktop click, *"Schedule Automatic Links."*
2. Next, in the window that opens, near the top, click Edit >> Preferences.
3. In the new small window that opens, you will need to enter your gateway's IP address, the port number, and the password.
4. In the "*Address*," textbox, type without quotes, "*127.0.0.1".*
5. In the "*Port"* textbox, type the port number you entered in the ircddbGateway Remote Tab Setup section.
6. In the "*Password"* textbox, type the password you entered in the ircddbGateway Remote Tab Setup section.
7. Press "*OK."*
8. Now close the Timer Control window.
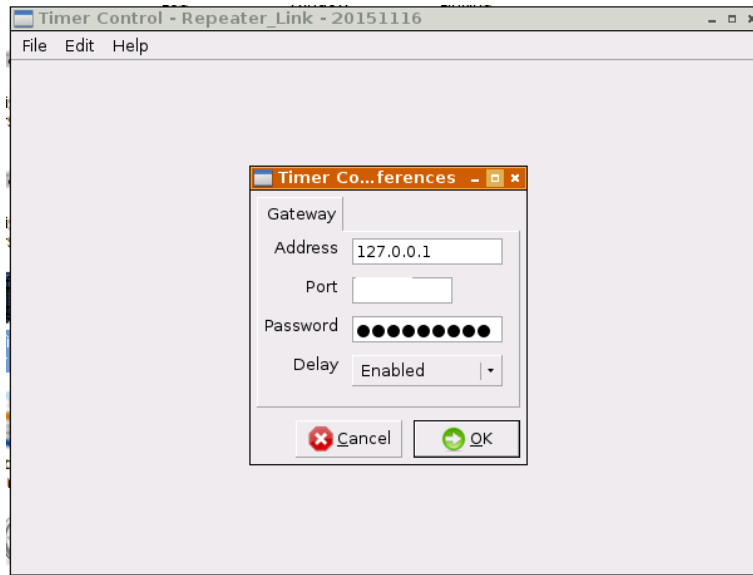
Setup Your Automatic Links

1. On the Pi desktop, click *"Schedule Automatic Links."*
2. Within a few seconds, a window should appear that looks similar to the picture below.
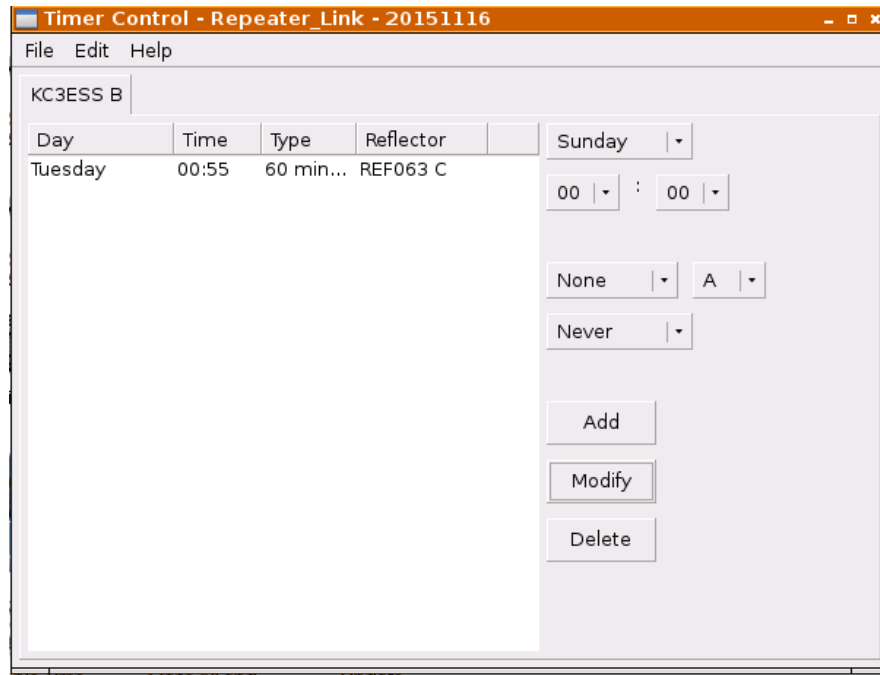
3. The left of this window will display any automatic links you've setup previously.
4. On the right side of this window, you can schedule, modify, or delete automatic links.
5. To setup a link, follow the steps below.
   a. Select the day from the dropdown box.
   b. Select the hour and minute in the second row of dropdown boxes. (Time is UTC)
   c. In the third row of dropdown boxes, select the reflector and it's module letter.
   d. In the fourth row of dropdown boxes, select the amount of time to stay connected to the link.
   e. Then click add.

Configure the Hardware

In this next section, Ted Leonard W3VG, talks about configuring the hardware for the DSTAR Repeater.

NOTE: You'll probably want to setup the hardware before the software.

Interest in DSTAR

About a year and a half ago DSTAR attracted my attention. There are no DSTAR repeaters anywhere near where I live in NW PA, so I was on my own. Some playing with a DVAP and Raspberry Pi generated further interest. Then I discovered many articles about home brewing DSTAR repeaters and lots of different software that would handle the internet and linking side of things, DSTAR is after all, open source.

I was fortunate to have "stuff" around as potential candidates for the project and had three analog FM repeaters on the air. The internet produced several articles on how various people did this. I bought a DVRPTR_V1 (Now discontinued – Use this board instead - http://www.mmdvm.com/) to handle the conversion of analog to digital and a Raspberry Pi computer handled the internet and linking side of things. An old Kenwood TKR820 UHF repeater was the first try---and it worked! At this point I must however mention that I am not a computer person let alone one with Linux skills. Tyler N3TDM is such a person and he joined the "process". The Kenwood worked great and got a few people interested, the downside is the repeater was at my house because I had good internet connectivity.

It wasn't long before it became apparent that a repeater needed to go to my excellently located repeater site to be of any real use. There is no internet at the site and on my pension there wasn't going to be. I had a UHF repeater at the site connected to AllStar by a link at my house and it worked fine--how could I do this with DSTAR?

Link to an Existing FM Repeater

A board called a DRC came to my attention from Advanced Repeater Systems that would enable conversion of an FM repeater to DSTAR. Now this will repeat only DSTAR data not analog FM and it has no internet connectivity all it does is repeat DSTAR conversations. It worked as advertised.

DRC Board & Info: http://arsrepeaters.com/D-Star_Repeater_Control.php

Most people contemplating such a project will already be familiar with the Repeater-Builder site and there is a new category there called "DMR and DSTAR Technical

Information" by Bob Meister WA1MIK (http://www.repeater-builder.com/dmr/dmr-index.html) one of the articles is "Interfacing Common Radios to DSTAR Controllers" and this is where it gets interesting. The Kenwood repeater I already had would be a bit anemic for mountaintop service and we still had a UHF MSF5000 repeater rescued from the county trash pile destined for the landfill and Bob covers using one of these wonderful repeaters for DSTAR in his article and emphasizes the need for a true FM repeater and access to the discriminator and direct input to the modulator for unprocessed audio in order to work with a digital mode such as DSTAR.

For anyone not familiar with Bob's articles he has a number of them on the MSF5000 and many other radios on the website and goes into great detail in all his articles and they have been of tremendous help to me and many others.

So now we have a repeater that happily passes DSTAR signals--how to get internet to the site so we can have full functionality in the world of DSTAR.

Send Internet to Remote Repeater Site

As previously mentioned there are many flavors of DSTAR software and we have settled on using the Maryland DSTAR image on a Raspberry Pi, one of the choices is Gateway which is just that, providing a Gateway/link to the repeater. We use the same DVRPTR_V1 on a Pi interfaced to the 9600 baud packet port available on most modern ham transceivers and it faithfully passes the audio and allows control of what reflectors or repeaters are linked to the system.

Setting the Deviation

Some thoughts on setting deviation for DSTAR. Most of the literature regarding deviation for DSTAR calls for 1.2 kHz deviation and while I have a nice HP service monitor the deviation on the MSF5000 with the DRC board was set by ear as I have found the various DSTAR radios are fussy about the level and I have always ended up with slightly more deviation than called for to keep them all happy. When adjusting the DRC MSF5000 levels I gave a long count on my ID5100 while someone else used my ID51A+ to listen for the signal while going through the adjustment of the DRC, on the first try nothing was decoded so the transmit polarity was flipped (a simple jumper on the DRC) the position of the adjustment pot was noted when it first decoded and continued adjustment until it was lost then the pot was set midway between the two points and it has been just fine. I guess it was done by the seat of the pants and all is well.

A big thanks to Bob Meister WA1MIK for all his work as published on the R-B site