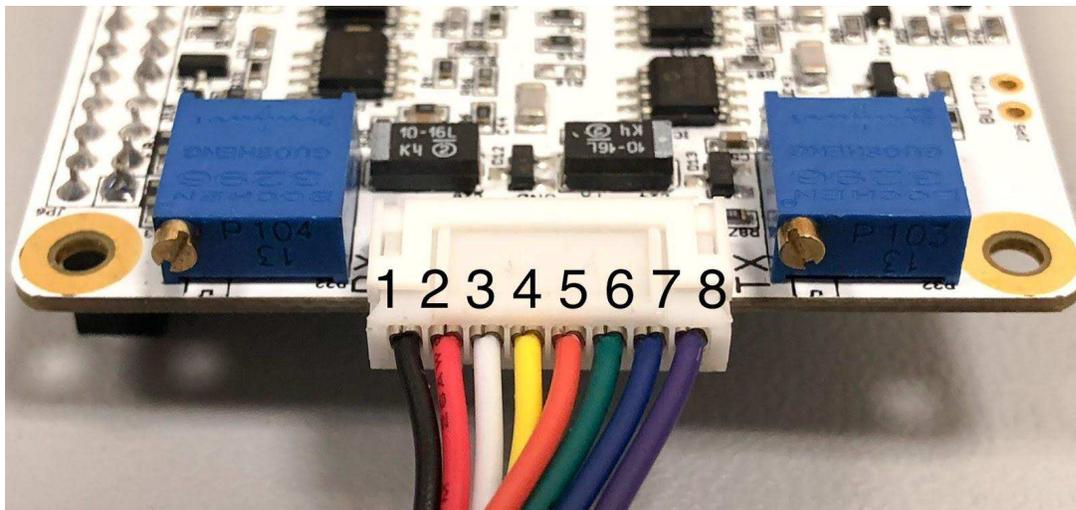


MMDVM-Pi rev. 1.0 board

- Designed for repeater and high power hotspot applications – connects to user supplied Raspberry Pi board and radio(s)
- Utilizes a high performance 32bit ARM processor (STM32F722) running at 180Mhz (Room for future upgrades)
- Fifth generation analogue filter design that provides an extremely low BER compared to previous generations
- 2 Multi-turn pots for fine RX and TX adjustments
- Onboard LEDs to show status and modes (PTT, COS, Power, D-Star, DMR, P25, Fusion, NXDN and POCSAG)
- Onboard LED to show receive signal level clipping
- Connection for Nextion LCD screen
- 8 pin JST Connector with pigtail wires
- Open source MMDVM firmware preloaded and easily upgraded by software

Wiring connections

Here is a picture of the 8 pin header with cable showing wire colors:



Here is a table of the pin numbers, names and wire colors:

Pin number	Signal name	Description	Wire color
1	CTRL	Control (output)	Black
2	COS/STAT1	Carrier sense (input)	Red
3	RX audio	Receive audio from radio (input)	White
4	Ground	Signal ground	Yellow
5	Ground	Signal ground	Orange
6	TX audio	Transmit audio to radio (output)	Green
7	PTT	Push to talk (enable transmit) (input)	Blue
8	RSSI	Received signal strength indicator (input)	Purple

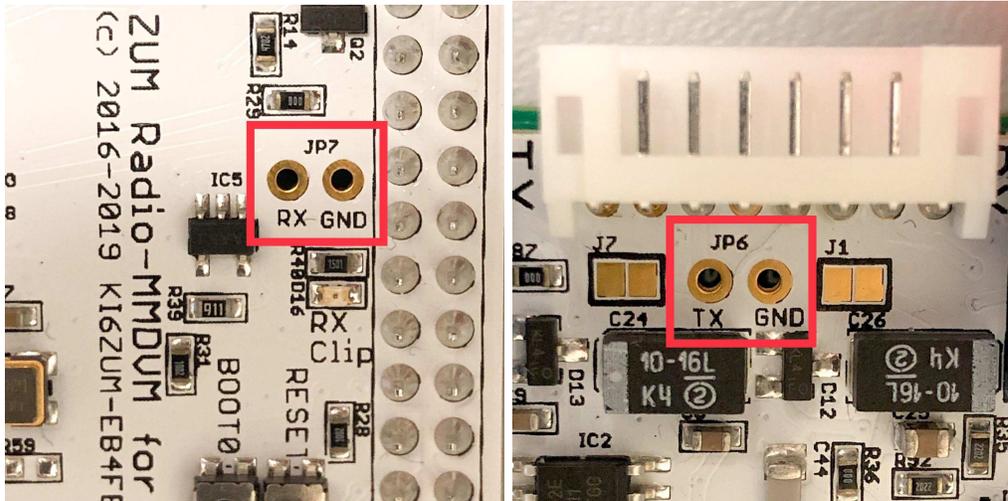
Here's an example radio's (FT-7900) pin connection via the mini DIN 6 connector on the rear of the radio:



Pin number	Signal name	Wire color	Mini DIN 6 connector
3	RX audio	White	9600 from radio
4	Signal ground	Yellow	GND
6	TX audio	Green	Audio to radio
7	PTT	Blue	PTT

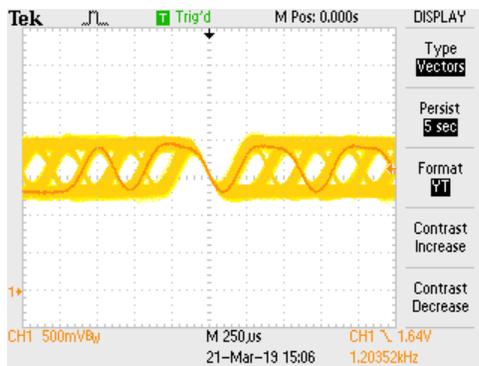
Adjusting signal levels (hardware)

Here is a picture of the test points on the PCB that can be used to look at the RX and TX signal levels on an oscilloscope. The signal on JP7 is the RX signal audio on the ADC input pin of the STM32 chip. JP6 has the TX signal audio on the pin going to the radio.

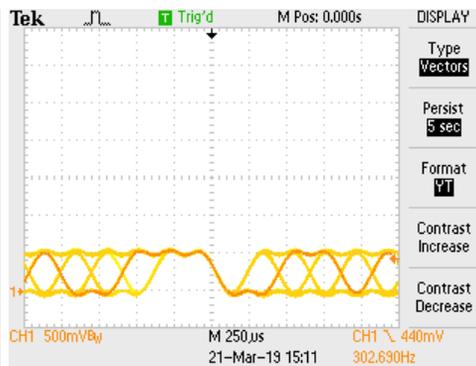


Here are some signal images captured on an Oscilloscope

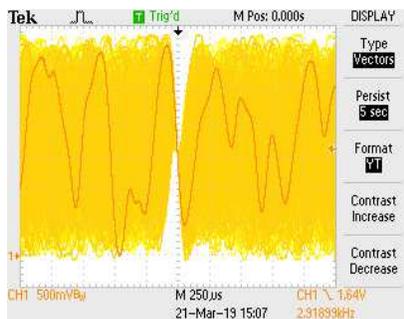
RX-(JP7)



TX-(JP6)



Noise without signal (JP7)



Here is a list of the jumpers on the board and their settings:

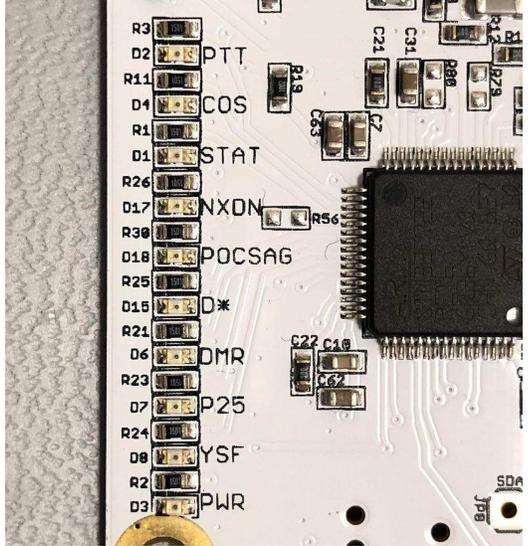
JUMPER DESCRIPTION TABLE

Bypass Pad	ON	OFF	Description
J1	X		Disable DC blocking capacitor on Radio RX end
J1		X	Enable DC blocking capacitor on Radio RX end
J2	X		Bypass 10K variable resistor
J2		X	Enable 10K variable resistor
J7	X		Disable DC blocking capacitor on Radio TX end
J7		X	Enable DC blocking capacitor on Radio TX end

Here is a list of the LEDs on the board and their settings:

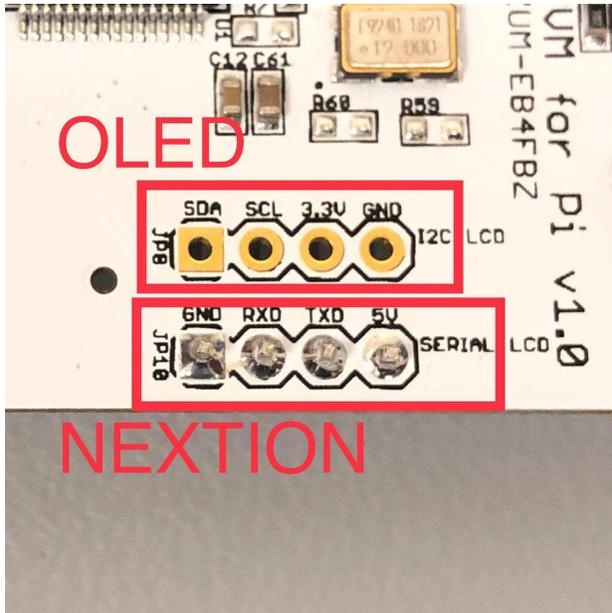
LED Description Table

LED NAME	Description
PTT	Board is transmitting
COS	Board is receiving
STAT	Status
D*	D-Star Enabled
PWR	Board is receiving power

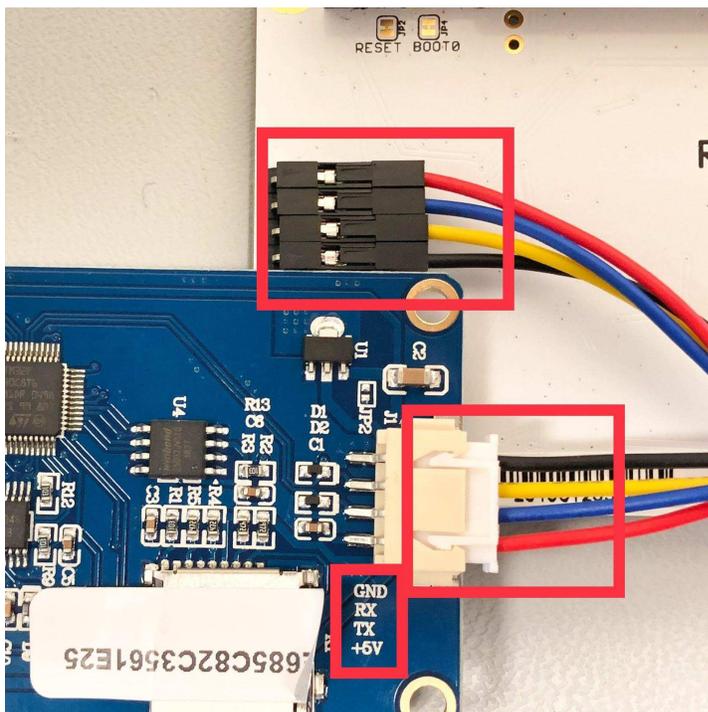


Adding an external display

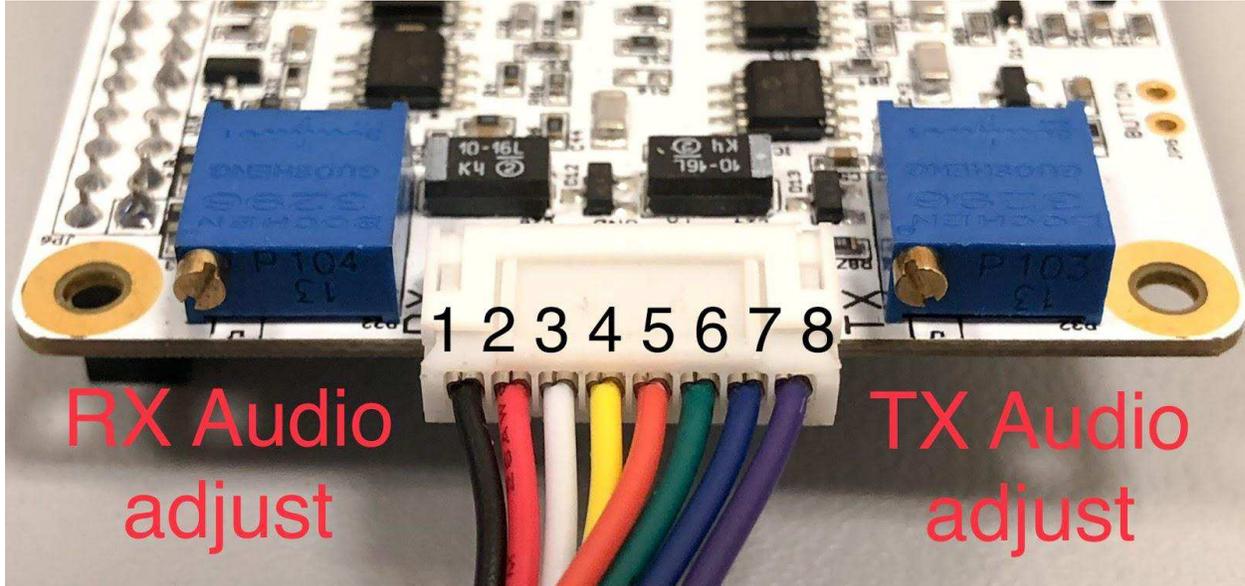
JP8 is where you can solder a header for connecting an OLED display. JP10 is where you can solder a header for connecting a NEXTION display.



For connecting a NEXTION display, make sure that each colored wire matches the connections shown in the image below.



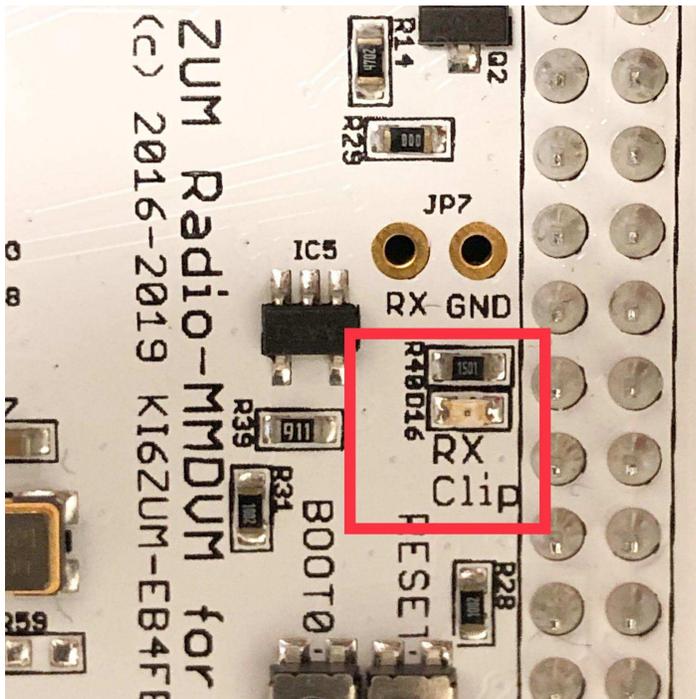
Here is a picture of the two trim pots used to adjust the signal levels of the RX and TX audio



Turning the RX audio trim pot counter-clockwise will increase the signal level, while turning the trim pot clockwise will decrease the signal level.

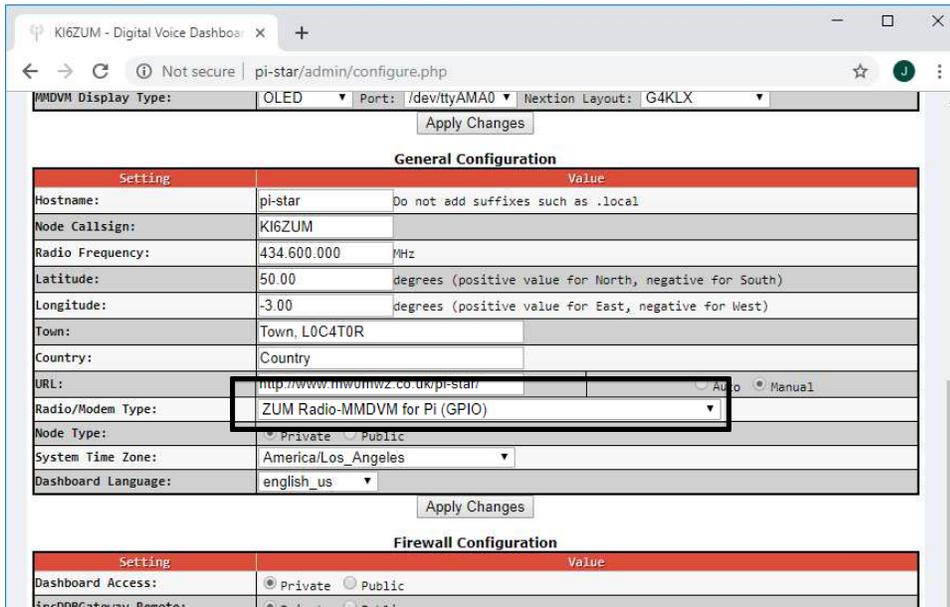
Turning the TX audio trim pot counter clockwise will decrease the signal level, while turning the trim pot clockwise will increase the signal level.

When there is no audio stream coming in, adjust the RX pot until the RX CLIP LED barely lights up.



Configuring Pi-Star (including adjusting signal levels via software)

From the “Configuration” menu, set the Radio/Modem type to “ZUM Radio-MMDVM for Pi (GPIO)” and then select Apply Changes.



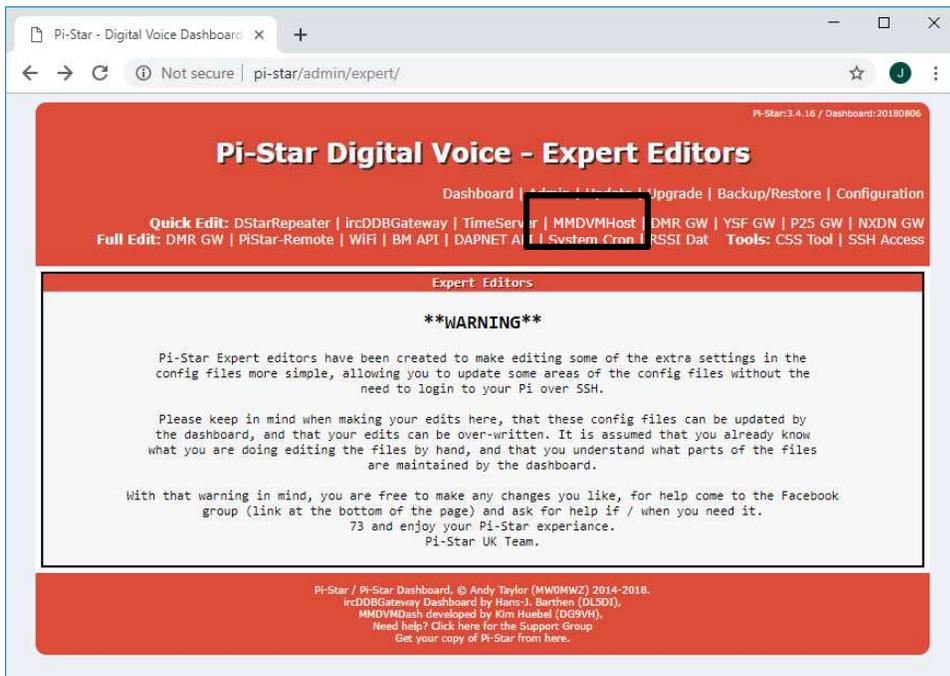
The screenshot shows the Pi-Star configuration interface. At the top, there are dropdown menus for MMDVM Display Type (OLED), Port (/dev/ttyAMA0), and Nextion Layout (G4KLX). Below these is an "Apply Changes" button. The main section is titled "General Configuration" and contains a table of settings:

Setting	Value
Hostname:	pi-star Do not add suffixes such as .local
Node Callsign:	KI6ZUM
Radio Frequency:	434.600.000 MHz
Latitude:	50.00 degrees (positive value for North, negative for South)
Longitude:	-3.00 degrees (positive value for East, negative for West)
Town:	Town, L0C4T0R
Country:	Country
URL:	http://www.m0wmw2.co.uk/pi-star <input type="radio"/> Auto <input checked="" type="radio"/> Manual
Radio/Modem Type:	ZUM Radio-MMDVM for Pi (GPIO)
Node Type:	<input checked="" type="radio"/> Private <input type="radio"/> Public
System Time Zone:	America/Los_Angeles
Dashboard Language:	english_us

Below the General Configuration section is another "Apply Changes" button. The next section is titled "Firewall Configuration" and contains a table of settings:

Setting	Value
Dashboard Access:	<input checked="" type="radio"/> Private <input type="radio"/> Public
ircDDBGateway Remote:	<input checked="" type="radio"/> Private <input type="radio"/> Public

Next select the “expert” menu, and then select “MMDVMHost”



The screenshot shows the Pi-Star Digital Voice - Expert Editors page. The page title is "Pi-Star Digital Voice - Expert Editors". Below the title is a navigation bar with links: Dashboard | Home | Update | Upgrade | Backup/Restore | Configuration. The main content area is titled "Expert Editors" and contains a warning message:

****WARNING****

Pi-Star Expert editors have been created to make editing some of the extra settings in the config files more simple, allowing you to update some areas of the config files without the need to login to your Pi over SSH.

Please keep in mind when making your edits here, that these config files can be updated by the dashboard, and that your edits can be over-written. It is assumed that you already know what you are doing editing the files by hand, and that you understand what parts of the files are maintained by the dashboard.

With that warning in mind, you are free to make any changes you like, for help come to the Facebook group (link at the bottom of the page) and ask for help if / when you need it.
73 and enjoy your Pi-Star experience.
Pi-Star UK Team.

At the bottom of the page, there is a footer with copyright information: Pi-Star / Pi-Star Dashboard, © Andy Taylor (M0WMW2) 2014-2018. ircDDBGateway Dashboard by Hans-J. Barben (DL3DJ), MMDVMDash developed by Kim Huebel (DG9VH). Need help? Click here for the Support Group. Get your copy of Pi-Star from here.

Scroll down to the “Modem” section to access the invert and level settings then select “Apply Changes”

Modem	
Port	/dev/ttyACM0
TXInvert	1
RXInvert	0
PTTInvert	0
TXDelay	100
RXOffset	0
TXOffset	0
DMRDelay	0
RXLevel	50
TXLevel	50
RXDCOffset	0
TXDCOffset	0
RFLevel	100
CwIdTXLevel	50
D-StarTXLevel	50
DMRTXLevel	50
YSFTXLevel	50
P25TXLevel	50
MXDNTXLevel	50
POCSAGTXLevel	50
RSSIMappingFile	/usr/local/etc/RSSI.dat
Trace	0
Debug	0

Scroll back up to the top of the page and select “Configuration”.

Pi-Star Digital Voice - Expert Editors

Dashboard | Admin | Update | Backup/Restore | **Configuration**

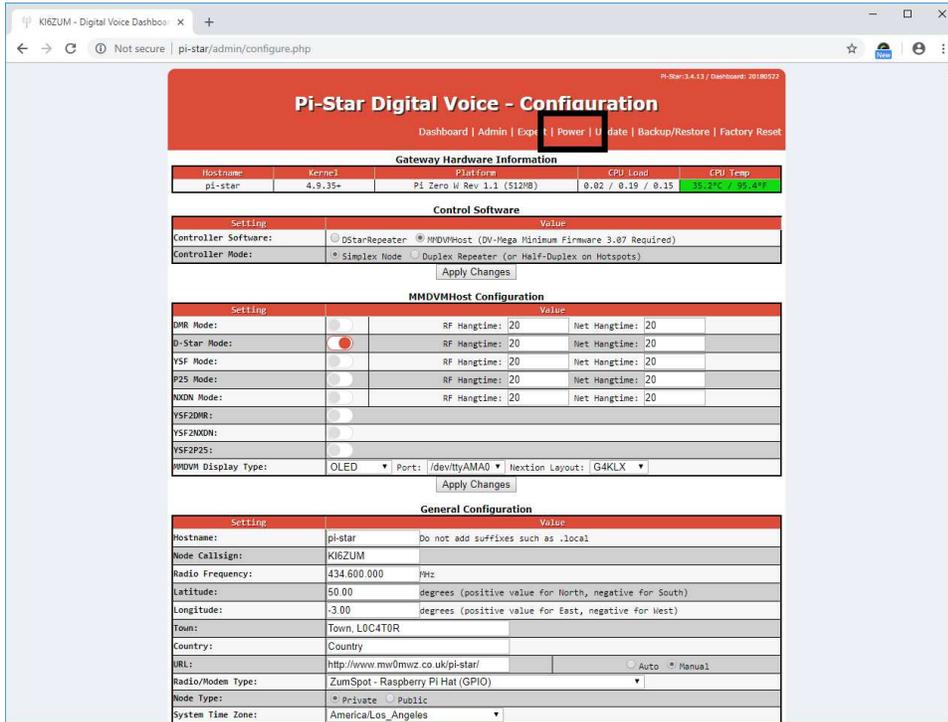
Quick Editors: DStarRepeater | ircDDBGateway | TimeServer | MMDVMHost | DMRGateway | YSFGateway | P25Gateway
Full Editors: DMRGateway | PiStar-Remote | WiFi Config | BM API Key | System Cron | RSSI Dat | Tools: SSH Access

General	
Callsign	KI6ZUM
Id	1234567
Timeout	240
Duplex	0
RfNodeHang	300
NetNodeHang	300
Display	OLED
Daemon	1

Info	
RXFrequency	434600000
TXFrequency	434600000
Power	1
Latitude	50.00
Longitude	-3.00
Height	0
Location	Town, LOC4TOR
Description	Country
URL	http://www.mw0mwz.co.uk/pi

Log	
DisplayLevel	0
FileLevel	2
FilePath	/var/log/pi-star
FileRoot	MMDVM

Next, select "Power"



Pi-Star Digital Voice - Configuration
Dashboard | Admin | **Power** | Update | Backup/Restore | Factory Reset

Hostname	Kernel	Platform	CPU Load	CPU Temp
pi-star	4.9.35+	Pi Zero W Rev 1.1 (512MB)	0.02 / 0.19 / 0.15	37.2 / 37.5 / 34.9

Control Software

Setting	Value
Controller Software:	<input type="radio"/> DStarRepeater <input checked="" type="radio"/> MMDVHost (DV-Mega Minimum Firmware 3.07 Required)
Controller Mode:	* Simplex Mode <input type="checkbox"/> Duplex Repeater (or Half-Duplex on Hotspots) <input type="checkbox"/>

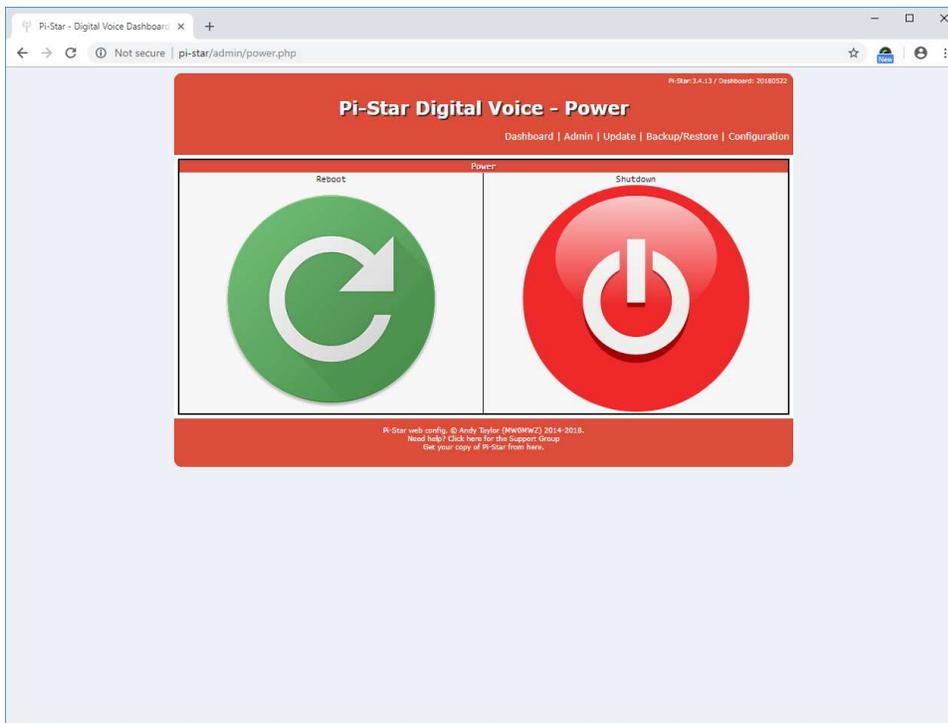
MMDVHost Configuration

Setting	Value
DMR Mode:	<input type="checkbox"/> RF Hangtime: 20 Net Hangtime: 20
D-Star Mode:	<input checked="" type="checkbox"/> RF Hangtime: 20 Net Hangtime: 20
YSF Mode:	<input type="checkbox"/> RF Hangtime: 20 Net Hangtime: 20
P25 Mode:	<input type="checkbox"/> RF Hangtime: 20 Net Hangtime: 20
NXDN Mode:	<input type="checkbox"/> RF Hangtime: 20 Net Hangtime: 20
YSF2DMR:	<input type="checkbox"/>
YSF2NXDN:	<input type="checkbox"/>
YSF2P25:	<input type="checkbox"/>
MMDVM Display Type:	OLED Port: /dev/ttyAMA0 Hexion Layout: G4K LX

General Configuration

Setting	Value
Hostname:	pi-star Do not add suffixes such as .local
Node Callsign:	K16ZUM
Radio Frequency:	434.600.000 MHz
Latitude:	50.00 degrees (positive value for North, negative for South)
Longitude:	-3.00 degrees (positive value for East, negative for West)
Town:	Town LOC4TOR
Country:	Country
URL:	http://www.m0mwz.co.uk/pi-star/ <input type="radio"/> Auto <input checked="" type="radio"/> Manual
Radio/Modem Type:	ZumSpot - Raspberry Pi Hat (GPIO)
Node Type:	* Private <input type="checkbox"/> Public <input type="checkbox"/>
System Time Zone:	America/Los_Angeles

Next, select "Reboot"



Pi-Star Digital Voice - Power
Dashboard | Admin | Update | Backup/Restore | Configuration

Reboot Power Shutdown

Pi-Star web config. © Andy Taylor (M0MWZ) 2014-2018.
Need help? Click here for the Support Group.
Get your copy of Pi-Star from here.

Building firmware

On Windows 10

- Install bash using these instructions: [bash-windows-10](#)
- Once bash is installed, install GCC for ARM. Open the command prompt and type in the following instructions.
 - `bash`
 - `sudo apt-get install gcc`
 - `sudo apt-get install make`
 - `sudo apt-get remove gcc-arm-none-eabi`
 - `sudo add-apt-repository ppa:team-gcc-arm-embedded/ppa`
 - `sudo apt-get install -y git gcc-arm-embedded=5-2015q4-1~trusty1`
- Make sure git is installed. If it isn't then use this command: `sudo apt-get install git`
- Get the latest source code from GitHub:
 - `git clone https://github.com/g4klx/MMDVM`
 - `cd MMDVM`
 - `git submodule init`
 - `git submodule update`
- Edit Config.h. Uncomment the line: `#define MODE_LEDS`
 - If you want to connect a Nextion display, then also uncomment the line:
`#define SERIAL_REPEATER`
- To start build run: `make pi-f722`
- Binaries will be under the `bin/` folder

On Ubuntu

- Follow the same instructions as **Windows 10** but skip the part about installing bash

On macOS

- First install Homebrew:
 - Open the Terminal
 - Paste this in and press Enter: `/usr/bin/ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install)"`
- Once Homebrew has been installed, run the following command: `brew install libusb autogen automake wget pkg-config cmake openocd`
- Install the ARM GCC toolchain:
 - Run the following command: `brew tap ArmMbed/homebrew-formulae`
 - Run the following command: `brew install arm-none-eabi-gcc`
- Get the latest source code from GitHub:

- *git clone* <https://github.com/g4klx/MMDVM>
- *cd MMDVM*
- *git submodule init*
- *git submodule update*
- Edit Config.h. Uncomment the line: *#define MODE_LEDS*
 - If you want to connect a Nextion display, then also uncomment the line: *#define SERIAL_REPEATER*
- To start build run: *make pi-f722*
- Binaries will be under the *bin/* folder

Flashing MMDVM-Pi

- Turn on Raspberry Pi
- Go to *Configuration->Expert->SSH Access*
- Login to pi-star
- Run command *rpi-rw*
- Transfer *bin/* folder from the previous section **Building Firmware** to the Raspberry Pi
 - On Windows, use WinSCP to transfer the folder
 - On Mac, you can use the Terminal by running the command: *scp -r ~/location_of_folder/bin pi-star@pi_star_ip_address:~/*
 - You will need to change *location_of_folder* to the directory of the */bin* folder and you will also need to change *pi_star_ip_address* to the IP address off the Raspberry Pi.
- Also transfer the *MMDVM-Pi_FW_Update.sh* script to the same directory on the Raspberry Pi
- From the SSH window of pi-star, navigate to the folder where the install script and *bin/* folder are in.
- Run command *./MMDVM-Pi_FW_Update.sh*
- The install script will take care of the flashing the MMDVM-Pi

On Windows 10

This is a web page with a good tutorial on how to use the *mmdvmcal* software tool and a spectrum analyzer to adjustment the signal levels for DMR:

<https://www.f5uui.net/en/installation-calibration-adjustment-tunning-mmdvm-mmdvmhost-raspberry-motorola-gm360/5/>

Another document describing the spectrum adjustment process:

<http://www.swedmr.se/wp-content/uploads/2017/08/Justering-av-repeater-med-MMDVM.pdf>

Support

Great video from W1MSG showing getting started with Pi-Star:

<https://www.youtube.com/watch?v=B5G4gYDdJeQ>

MMDVM Yahoo group:

<https://groups.yahoo.com/neo/groups/mmdvm/conversations/messages>

Pi-Star support forum:

<https://forum.pistar.uk/>

Pi-Star Facebook support group:

<https://www.facebook.com/groups/pistar/>

Pi-Star Wiki:

<http://wiki.pistar.uk>

A web page describing the setup of an MMDVM repeater:

<https://sadirigitalradio.com/digital-radio-how-tos/make-mmdvm-digital-repeater/>