# ***Setting Up the MTR2k Repeater with AllStar and MMDVM Mixed-Mode***

A guide by Larry K. Aycock, N6LKA

This is a long writeup, but hopefully has enough details for anyone looking for information on how to add AllStar and digital mixed-mode capabilities to a MTR2k repeater (you've been warned.). For a long time, I have wanted to setup a repeater that utilizes Allstar for analog and has digital mixed-mode (MMDVM) capabilities. A friend of mine, Patrick (AD5MT), has a MTR2k repeater with a MMDVM card making it a mixed-mode repeater, and had suggested I purchase a MTR2k, as it is a great repeater. I was hesitant for a long time to purchase a MTR2k because I was not sure it would do everything I wanted. However, the price of an MTR2k is hard to beat compared to other repeaters, and they are built very well. I finally purchased one to experiment with and try to get everything setup and working the way I want it. I was successful, and I now have my MTR2k in operation as a digital mixed mode repeater, using Pi-Star to control the MMDVM functions for digital modes, and my HamVOIP node on a Raspberry Pi4b controlling the analog functions.

Let me clarify that this is a mixed-mode repeater, not a cross-mode bridged repeater. I have had a few people ask me questions about my setup thinking they could talk on DMR through AllStar or some other cross-mode function. That is possible through another setup but is not the topic of this writeup. This is a mixed-mode repeater setup, meaning that the user can use the repeater in analog, DMR, YSF, D-Star, P-25, or NXDN, and the repeater will repeat the signal in the received mode. Since it is using a MMDVM modem, each of the respective digital modes will also connect to their respective broader networks, such as Brandmeister for DMR.

Below is a list of the equipment I am using, and the configuration setup I have with explanations to make it all work together (not all equipment is listed. E.g. duplexer, pre-amp, filters, antenna, etc.). I have seen a few others on various sites looking for this same type of setup, asking questions about how to add AllStar and mixed-mode capabilities to a MTR2k repeater. I hope posting this information helps others looking to use this setup on their repeater.

**Equipment:**

- Motorola MTR2000 Repeater – There are VHF and UHF versions of this repeater. I have the 40w UHF 403-470 split repeater. I purchased mine at https://used-radios.com/.
- TTN5067 Wireline Board installed in top slot in the repeater – some of these repeaters come with this board already installed. If not, you will need to purchase one to install. This setup will not work without a Wireline board installed. This board must also be installed in the top slot of the MTR2k.
- Raspberry Pi4b running the HamVOIP AllStar. You can purchase these very cheap on Amazon.
- DMK Engineering URIxB – Needed to interface the MTR2k to the Rpi4b. http://www.dmkeng.com/Products.htm
- I made a custom cable to connect the MTR2k to the URIxB. I will go into more details on these connections later. It is easier to make the cable if you solder onto a 96-pin DIN connector that will connect to the 96-pin on the back of the MTR2k. You can find the connectors on www.ebay.com
- STM32-DVM-MTR2k MMDVM Board for MTR2k repeaters. Cort Buffington makes an excellent MMDVM board that fits into one of the available accessory board slots on the front of the MTR2k. These boards seamlessly integrate and work well with the repeater to add digital modes with ease. I purchased mine from Cort and the folks at KS-DMR, but I believe Repeater Builder also makes a similar version of these boards (Cort's design) and sells them.
  http://ks-dmr.net/stm32-dvm-mtr2k-information/

## Setup MTR2k with STM32-DVM-MTR2k as a Mixed-Mode Repeater:

The first step for me was to install the MTR2k and get it working as an analog repeater. I then added the STM32 board and followed the instructions on KS-DMR to setup the digital capabilities for mixed-mode. Note: in this setup, you must have the repeater in “Repeater Operation” mode, not “Base Operation.” Later on, when adding in AllStar, we will change it to “Base Operation.”
Below are links to the articles on KS-DMR website on setting up the STM32 board.

- Programming an MTR2000 for the STM32-DVM-MTR2K – This is the initial setup of the MTR2k to work with the STM32.
- MTR2000 and STM32-DVM-MTR2K: Analog + Digital, Playing Nice Together – This article shows you how to make sure analog doesn’t hijack digital, and vice-versa. You can ignore this for now as I will go over this in more detail later.
- Configuring MMDVM for the MTR2000 and STM32-DVM-MTR2K – This is also something you can skip for now and wait until you have the final setup to make these adjustments.

Note: The STM32 comes preloaded with MMDVM software. I chose not to use that software and installed a new SD card with Pi-Star because I am more familiar with Pi-Star, and I like the web interface dashboard it offers. Use whichever you like, it does not matter.

## Custom Cable Connection between the MTR2k and URIxB:

Once you get the repeater working as a mixed-mode repeater with the STM32 board, then it is time to change things up and add on the AllStar node to control the analog side of the repeater. I already had my AllStar node running on the Rpi4 and setup for a repeater because I purchased the MTR2k to replace two CDM1550’s I was using in my repeater setup. I assume you are familiar with AllStar and know how to install and complete the basic setup of the node.

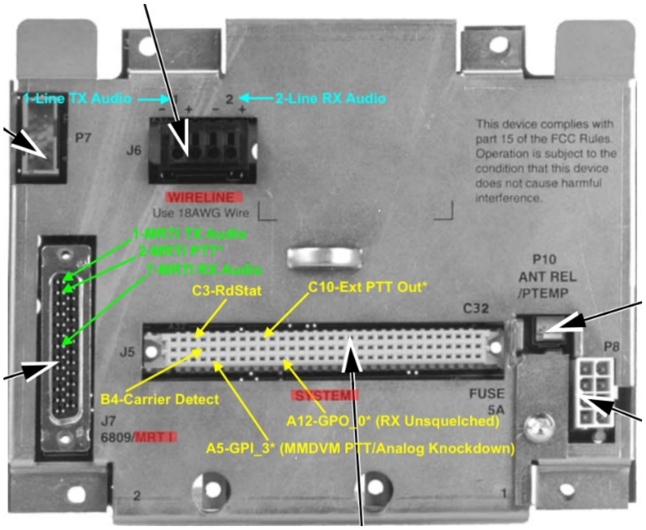
I again used the guide from KS-DMR to help walk me through this setup. However, I found there were certain areas of the article that needed a bit more detail and explanation, at least for me. 😊 I would recommend you read the article listed below before continuing.
STM32-DVM-MTR2K: MMDVM and an External Analog Controller

Although the article describes connecting the MTR2k, with the STM32 board installed, to an external analog controller, such as the ARCOM RC-210 or others, we are going to connect it to the AllStar node to work as a single port repeater controller.

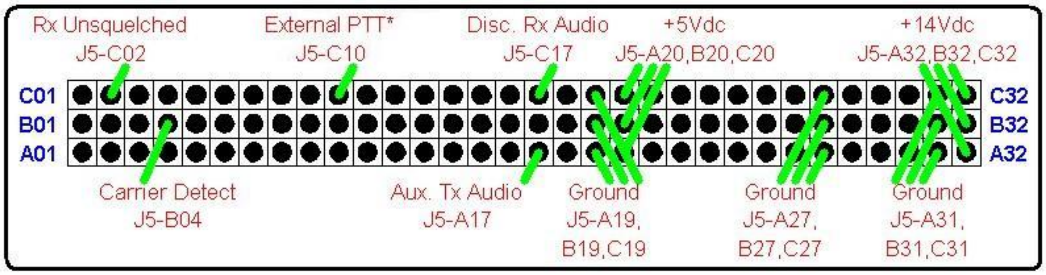
Below I will discuss how to connect the AllStar node to the MTR2k repeater. In the next section, I will discuss the programming changes to the RSS and AllStar.

As I mentioned before, I built a custom cable to connect the DB25-pin connector on the URIxB to the 96-pin connector, and Wireline connector on the rear of the MTR2k. Below pictures of the rear panel, and pin diagrams of the connectors along with a table outlining the wire pinout mapping of the custom cable I created.

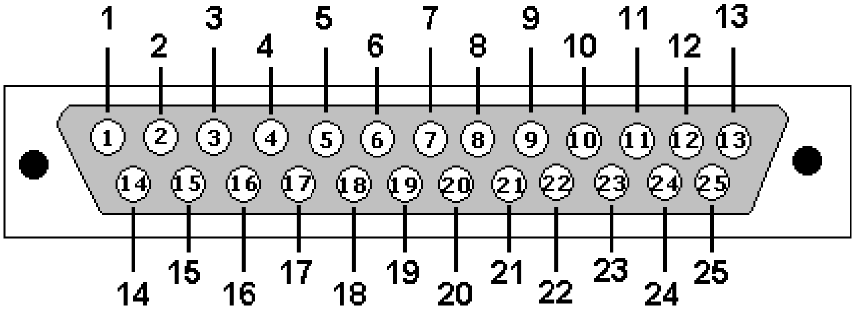
Tip: Tin the tips of the four wires that will plug into the Wireline connector with solder. It will make it easy to push the wires into the Wireline connector on the MTR2k. To disconnect, use a straightened paperclip to push into the square hole above each wire to release while gently pulling the wire out.


1-Line TX Audio
1
2
2-Line RX Audio
- + - +
P7
J6
WIRELINE
Use 18AWG Wire
This device complies with part 15 of the FCC Rules. Operation is subject to the condition that this device does not cause harmful interference.
1-MRTI TX Audio
2-MRTI PTT*
7-MRTI RX Audio
C3-RdStat
C10-Ext PTT Out*
P10
ANT REL
/PTEMP
C32
J5
P8
B4-Carrier Detect
SYSTEM
FUSE
5A
J7
6809/MRT I
A12-GPO_0* (RX Unsquelched)
A5-GPI_3* (MMDVM PTT/Analog Knockdown)
2
1

*This image shows the connections on the rear of the MTR2k. We will not use all the connections listed.*



*This image is looking at the 96-pin connector from the rear of the repeater. Again, we will not use all these connections. This image is for reference only.*



*This is the pinout of the DB-25 Male connector with the pins facing you. This connector will plug into the URIxB.*

## *Custom Cable Pinout Mapping*

| *MTR2k Connections* | | | | | *URIxB DB25-Pin* | |
|---|---|---|---|---|---|---|
| **Wireline Pin** | **Description** | **96-Pin** | **Description** | **Wire Color** | **DB-25 Male Pin** | **Description** |
| | | C5 | GPI_4 (WCI) - Programmed as PTT Key from WL with WC | White/Orange | 1 | PTT |
| | | A12 | RX Unsquelched programmed with WC | Orange | 7 | CTCSS_DET |
| | | C19 | Ground | Brown | 13 | GND |
| 2- | RX Audio Output | | | White/Blue | 19 | GND |
| 1- | TX Audio Input | | | White/Green | 20 | GND |
| 2+ | RX Audio Output | | | Blue | 21 | MIC_AC - Audio Input |
| 1+ | TX Audio Input | | | Green | 22 | Left_Out - AC Coupled Audio Output |

***Important*** - This is the pinout mapping I created and the connections I used to build my custom cable.

***PTT Connection and PL tone on TX:***

You may notice from the pictures above that there is an external PTT connection on pin C10. You may be wondering why I am using C5 instead of C10 for my PTT. This is an excellent question that requires a bit more explaining and has to do with using the PL of the MTR2k. It is documented on the Repeater Builder site that if you use the external PTT with Wireline, the repeater will not encode PL (CTCSS/DCS) when transmitting. I totally missed this when I read the articles and did not clearly understand what they were trying to say. It also seems like this important piece of information was buried in a table of other stuff, and not specifically called out. Now that I understand, I went back and re-read the articles, and it makes sense, now, although it could have been more clearly explained. I spent many frustrating hours/days trying to figure out why everything was working, except the PL encode on my setup. Finally, I reached out to Cort at KS-DMR for help via email. In his reply, he mentioned something in a way that clicked with me and helped me to understand. In his article, he suggested using an available GPI pin, and programming it to key Wireline. Doing it this way allows you to program a Wildcard to key Wireline which will encode PL on TX. Also, the STM32 board uses the external PTT, and it probably is not a good idea to have the AllStar node signaling PTT on a pin already used by the STM32. As such, if you want to use PL from the repeater, like I did, then you cannot use the external PTT connection on C10. I chose to use C5 which is GPI_4 and programmed it using a Wildcard table to Key from Wireline. I will go over programming the Wildcards in more detail later. I also used A12 as my RX Unsquelched and programed it using a Wildcard instead of using C2. This is needed so that you can program the MMDVM and analog side to play nice together using Wildcard settings so neither mode overrides the other.

As you can see from the table above, you only need to connect three wires to the 96-pin connector. The other four wires connect to the Wireline connector. However, there are other pins on the 96-pin connector that I used, such as the 5vdc and 12vdc pins. I added leads off these to power a GPS-DO for a reference signal input, and a lead to power the Rpi4 for the AllStar node. This allows me to power everything directly from the MTR2k. Below is the pinout I used for the power connections from the 96-pin connector in case you want to do the same, but it is not required.

Note: you will need a DC-DC converter to step down the power from 12v to 5v to power the Rpi4. I did this because the amp requirement of the Rpi4 is finicky, and I wanted to make sure I have enough amps to power it without issue. Also, the GPSDO only needs 5v and has a very low amp draw. It made sense to me to have the two devices on different power outputs.

| Power Connections for Rpi and GPSDO | |
|---|---|
| **96-Pin** | **Description** |
| C32 | +14V DC - Rpi |
| C31 | Ground - Rpi |
| A20 | +5V DC - GPSDO |
| A19 | Ground - GPSDO |

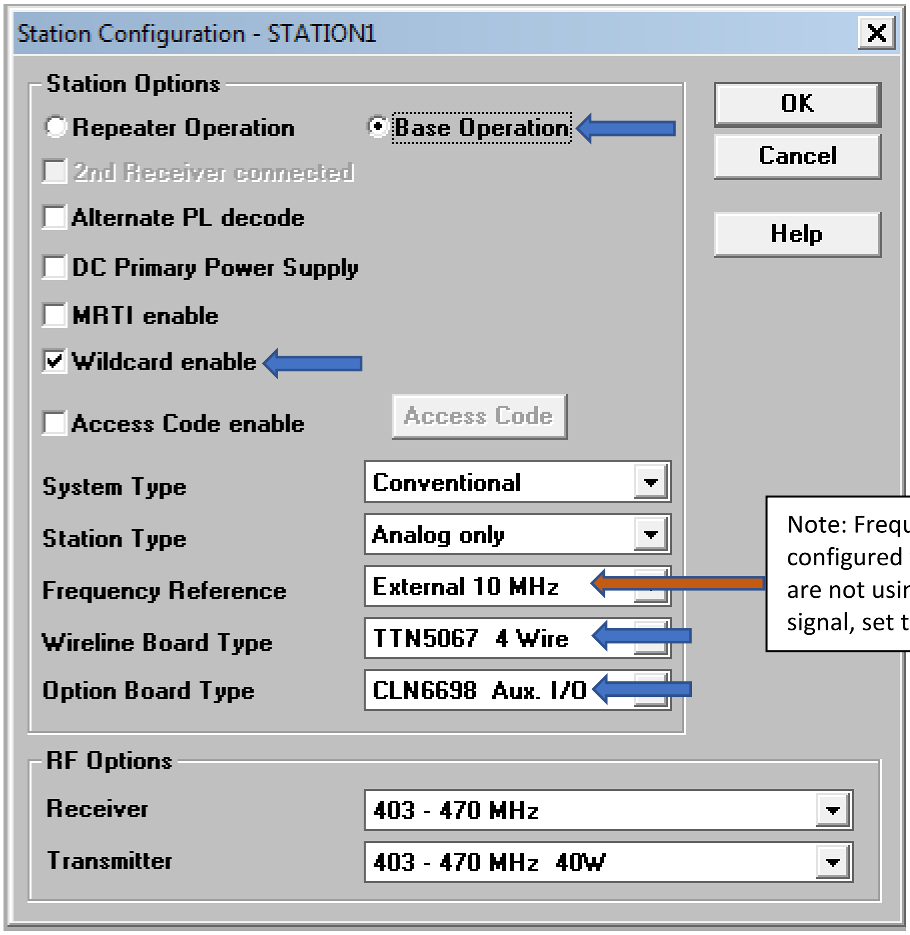
Take your time to build the cable and correctly match the pins from the 96-pin connector on the MTR2k to the DB-25 male which will connect to the URIxB.

# Configuring the MTR2k (with STM32) to work with an External Analog Controller (AllStar):

## ***Station Configuration Screen:***

Now, with everything connected properly, we must make changes to the RSS configuration of the MTR2k. You must put the repeater in "Base Operation" instead of "Repeater Operation" to disable the internal controller. It is necessary to tell the station which Wireline card is installed, and whether it is to operate in 2 wire or 4 wire. It is important that you select a 4-wire audio board. The Option Board Type should already be selected as the CLN6698 AUX I/O, and Wildcard enable as part of using the STM32-DVM-MTR2K.

- Base Operation
- Wildcard Enable
- Wireline Board Type = TTN50xx 4-wire
- Option Board Type = CLN6698 Aux. I/O



Station Configuration - STATION1

Station Options

○ Repeater Operation ● Base Operation

□ 2nd Receiver connected

□ Alternate PL decode

□ DC Primary Power Supply

□ MRTI enable

☑ Wildcard enable

□ Access Code enable

| | |
|---|---|
| System Type | Conventional |
| Station Type | Analog only |
| Frequency Reference | External 10 MHz |
| Wireline Board Type | TTN5067 4 Wire |
| Option Board Type | CLN6698 Aux. I/O |

RF Options

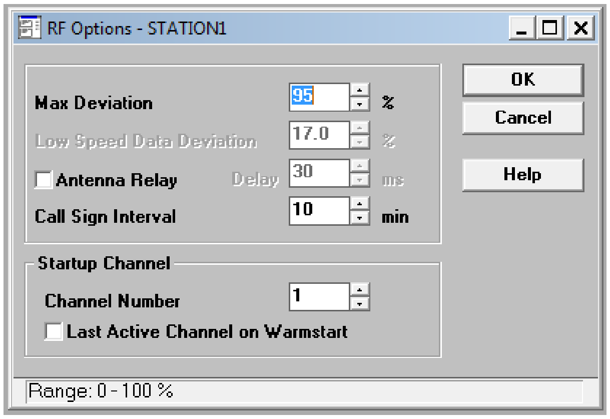
| | |
|---|---|
| Receiver | 403 - 470 MHz |
| Transmitter | 403 - 470 MHz 40W |

Note: Frequency Reference is configured for external 10 MHz. If you are not using a 10 MHz reference input signal, set this to "Internal Standard."

*Station Configuration showing Base Operation, Wildcard enabled, and a TTN50xx Wireline card installed, and configured for 4-wire audio.*

***RF Options Screen:***

There is not much to set on this screen, but I am including it here as a reference for how I have mine configured.  If you choose to use the MTR2k CW ID, you must set the Call Sign Interval.



RF Options - STATION1

Max Deviation 95 %

Low Speed Data Deviation 17.0 %

□ Antenna Relay Delay 30 ms

Call Sign Interval 10 min

Startup Channel

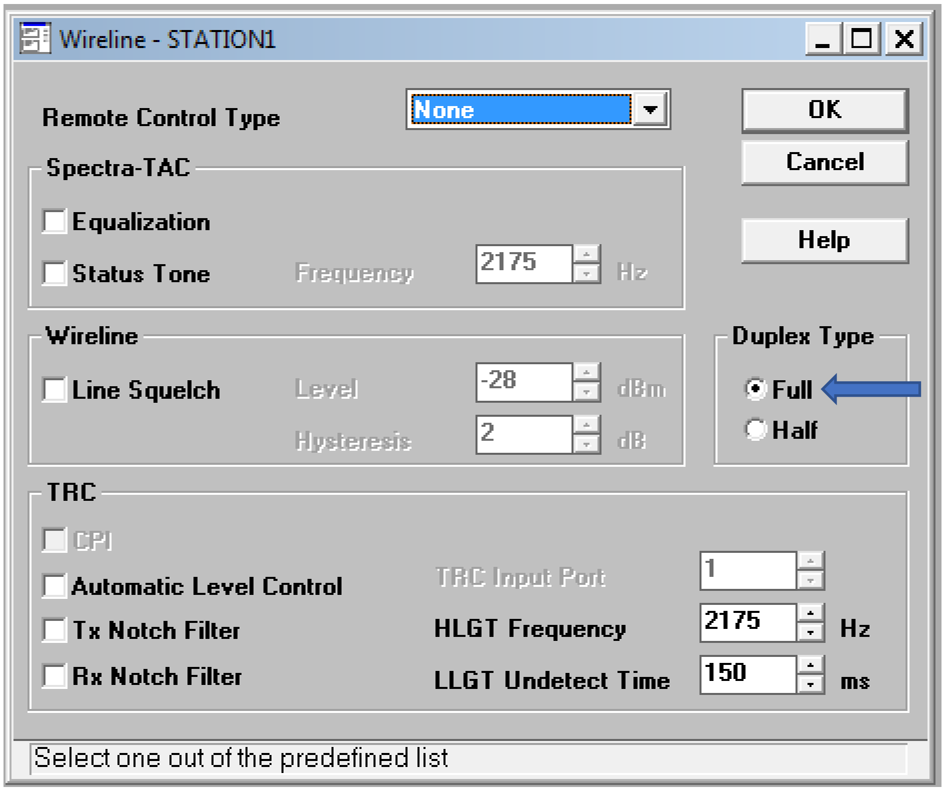
Channel Number 1

□ Last Active Channel on Warmstart

Range: 0 - 100 %

***Wireline Screen:***

The wireline Duplex Type must be set to “Full”, otherwise the RX audio will be muted (wireline and MRTI) when the transmitter is keyed.



Wireline - STATION1

Remote Control Type None

Spectra-TAC

□ Equalization

□ Status Tone Frequency 2175 Hz

Wireline

□ Line Squelch Level -28 dBm

Hysteresis 2 dB

Duplex Type

● Full

○ Half

TRC

□ CPI

□ Automatic Level Control TRC Input Port 1

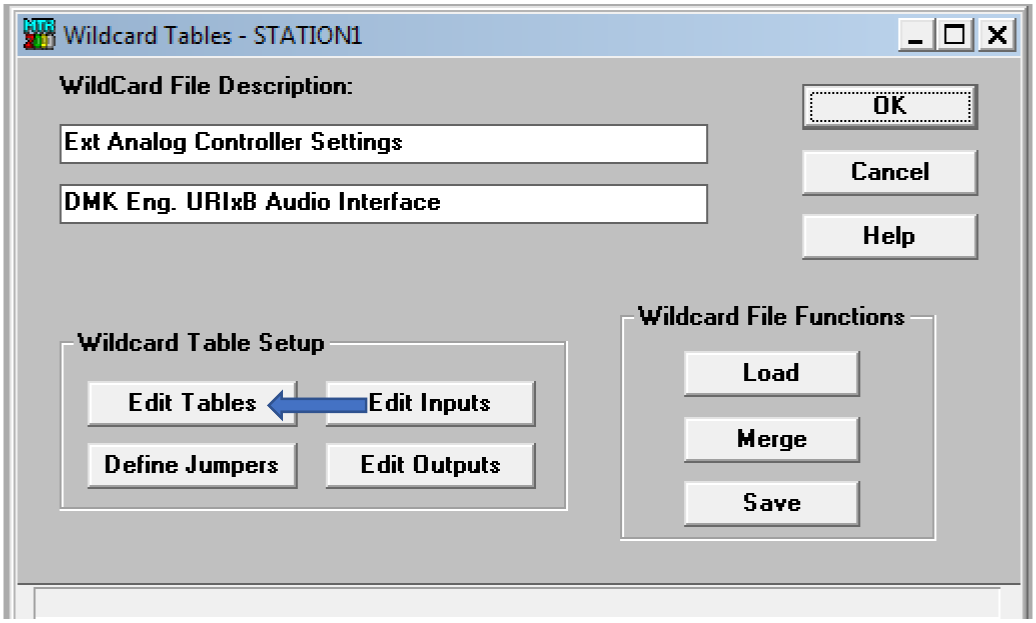
□ Tx Notch Filter HLGT Frequency 2175 Hz

□ Rx Notch Filter LLGT Undetect Time 150 ms

Select one out of the predefined list

***Wildcard Settings:***

In this section, I will show the setting I use for my Wildcard tables, and explain what each one is doing. Note that in order to use Wildcard tables, you must have a AUX I/O board installed to enable this function. Luckily, the features of the AUX I/O board have been built into the STM32-DVM board, allowing us to enable and use Wildcard tables. From the Personality menu, select Wildcard, and the following window should appear. Click the “Edit Tables” button.



Wildcard Tables - STATION1

WildCard File Description:

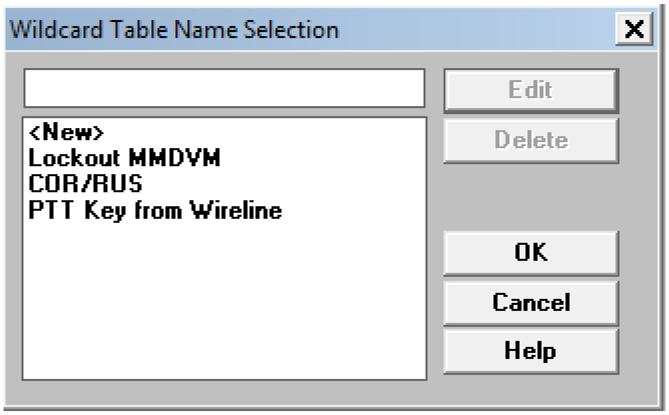
Ext Analog Controller Settings

DMK Eng. URIxB Audio Interface

Note: Only use the Edit Tables button. We do not need to edit inputs/outputs or define jumpers.

After clicking the “Edit Tables” button, the following screen will appear. This screen shows which Wildcard tables you currently have created. As you can see from the picture below, I have created three tables.

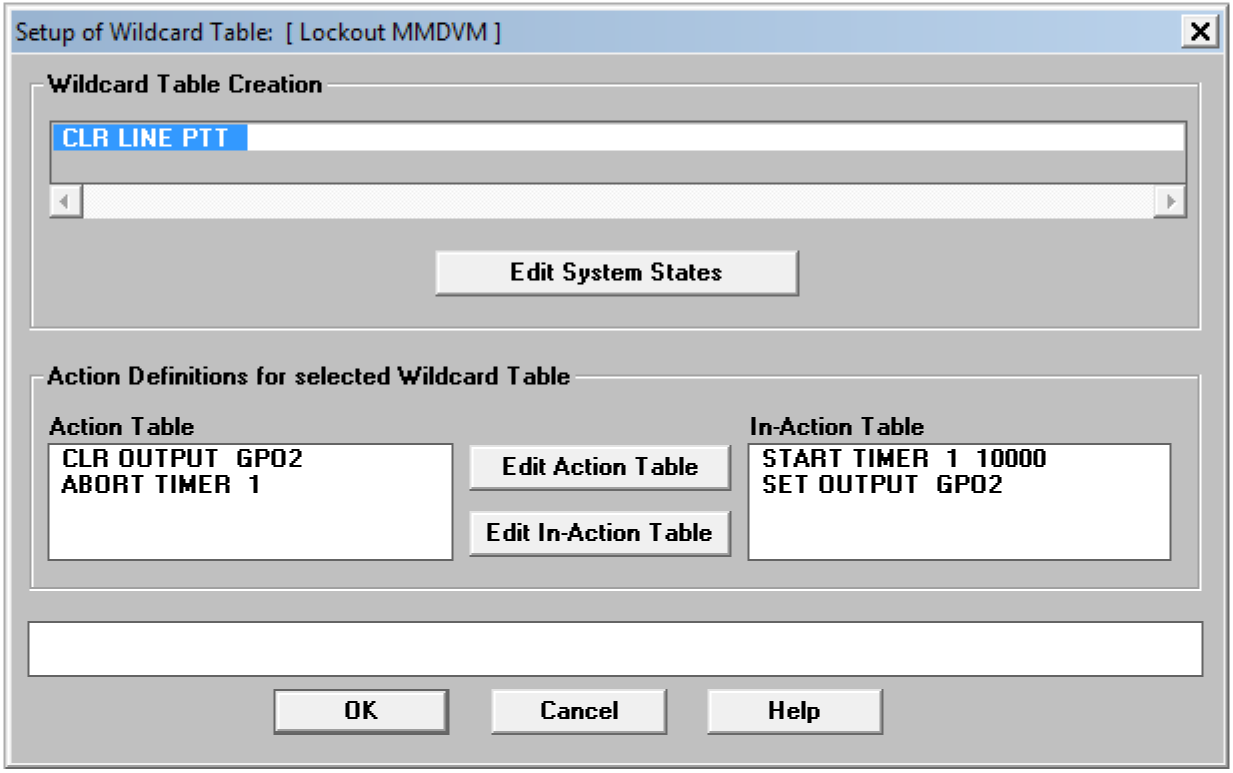
- Lockout MMDVM – used to lockout all digital modes when analog is in use.
- COR/RUS – used to toggle AllStar when there is a signal present.
- PTT Key from Wireline – Used to key/dekey the repeater using Wireline from AllStar.



Wildcard Table Name Selection

<New>
Lockout MMDVM
COR/RUS
PTT Key from Wireline

**Lockout MMDVM Wildcard Table:** Used to lockout all digital transmissions from the STM32 board while analog is in use. Note: the original article from Cort on KS-DMR.net has the action table with the abort timer on the first line, then the clear output for GPO2 on the second line. I found that this did not work. By switching the order to clear the output first, then abort the timer, the lockout function works correctly.



Setup of Wildcard Table: [ Lockout MMDVM ]

Wildcard Table Creation

CLR LINE PTT

Action Definitions for selected Wildcard Table

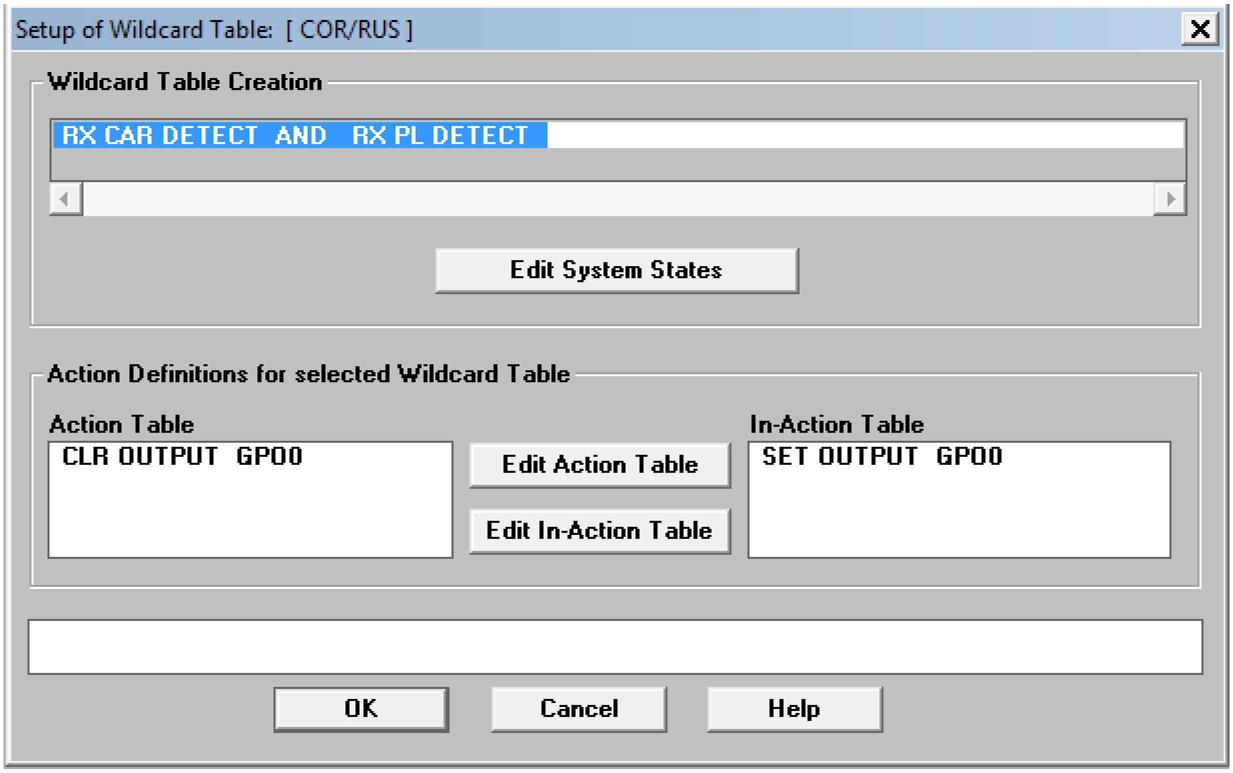
Action Table

```
CLR OUTPUT  GPO2
ABORT TIMER  1
```

In-Action Table

```
START TIMER  1  10000
SET OUTPUT  GPO2
```

**COR/RUS Wildcard Table:** Used to toggle the URIxB/Allstar when there is a RX signal present. The signal will activate when there is a carrier and the correct PL is detected on RX.



Setup of Wildcard Table: [ COR/RUS ]

Wildcard Table Creation

RX CAR DETECT  AND   RX PL DETECT

Action Definitions for selected Wildcard Table

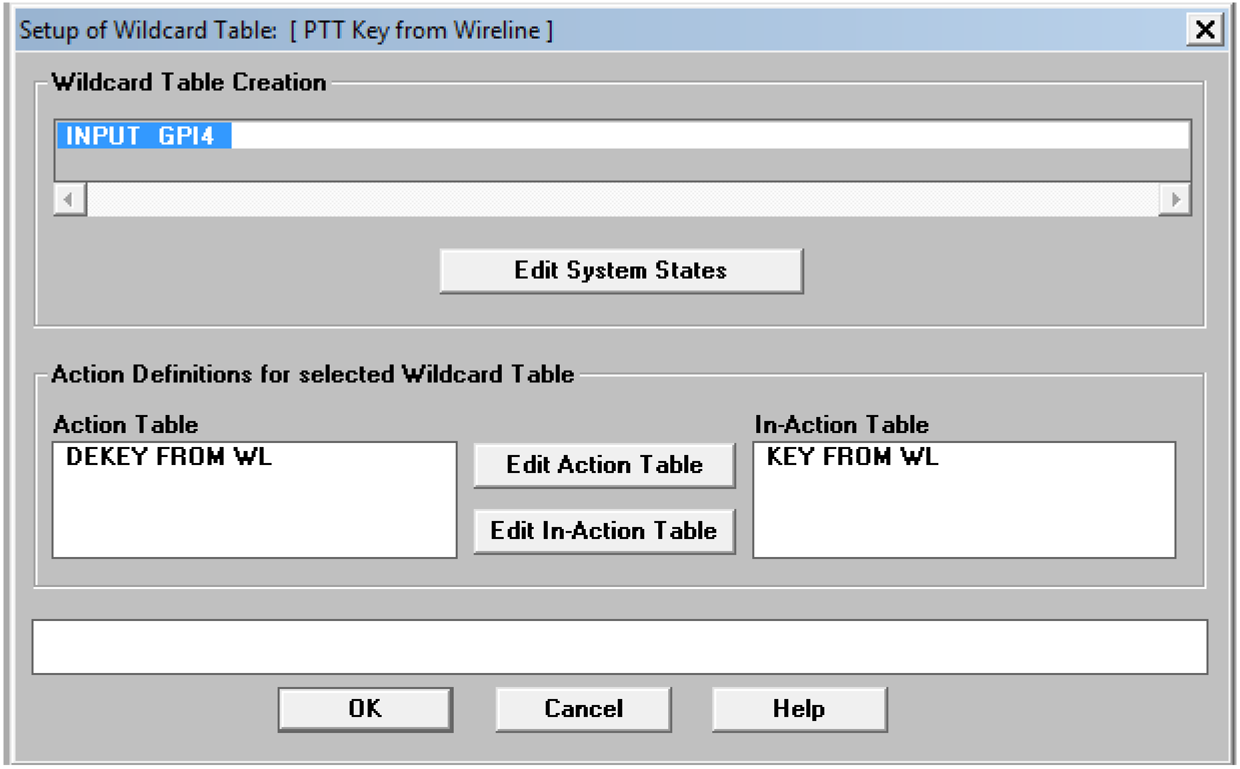
Action Table

```
CLR OUTPUT  GPO0
```

In-Action Table

```
SET OUTPUT  GPO0
```

**PTT key from Wireline Wildcard Table:** Used to key/unkey the repeater TX from AllStar\Wireline using GPI4. By using this wildcard table to key/unkey, it enables the use of PL with the TX.



Setup of Wildcard Table: [ PTT Key from Wireline ]

Wildcard Table Creation

INPUT GPI4

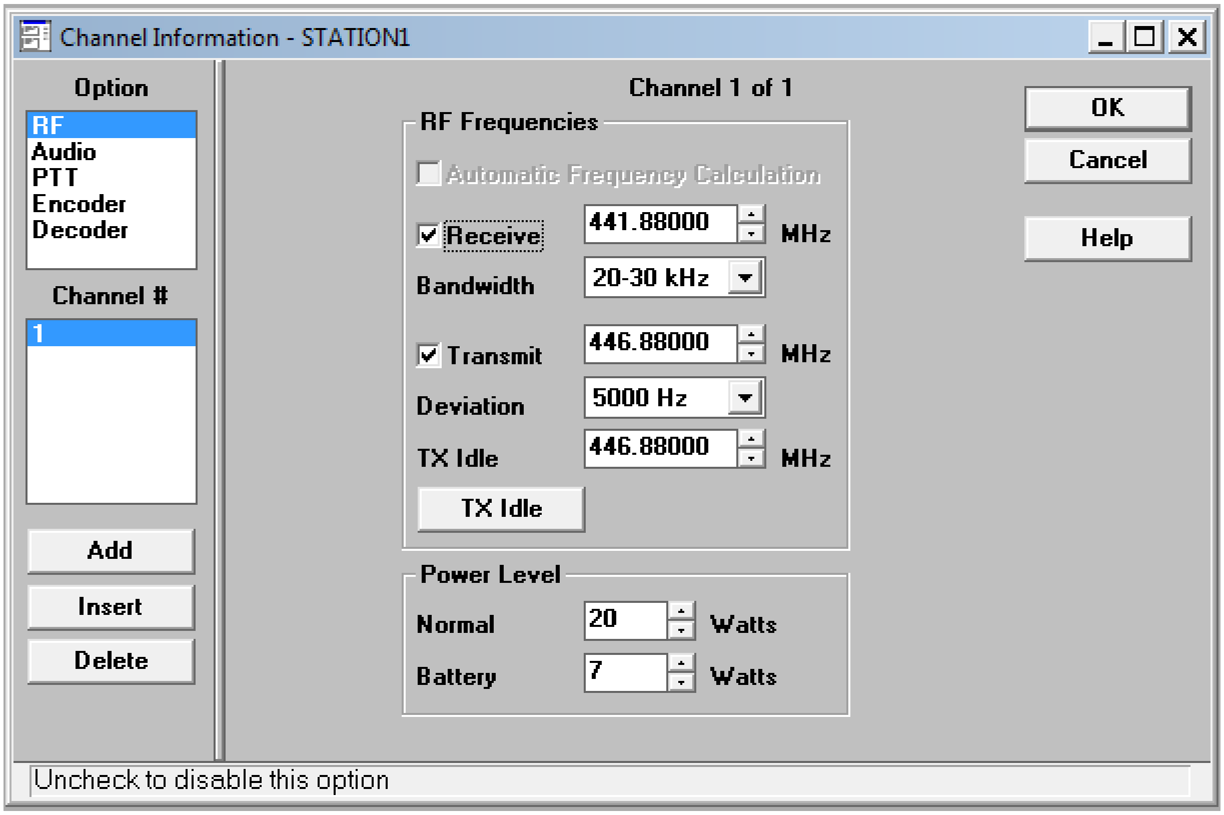
Action Definitions for selected Wildcard Table

Action Table: DEKEY FROM WL

In-Action Table: KEY FROM WL

## ***Channel Information Screens:***

**RF:** Set your input/output frequencies and output power on this screen.



Channel Information - STATION1

Option: RF, Audio, PTT, Encoder, Decoder

Channel #: 1

Channel 1 of 1

RF Frequencies

□ Automatic Frequency Calculation

☑ Receive 441.88000 MHz

Bandwidth 20-30 kHz

☑ Transmit 446.88000 MHz

Deviation 5000 Hz

TX Idle 446.88000 MHz

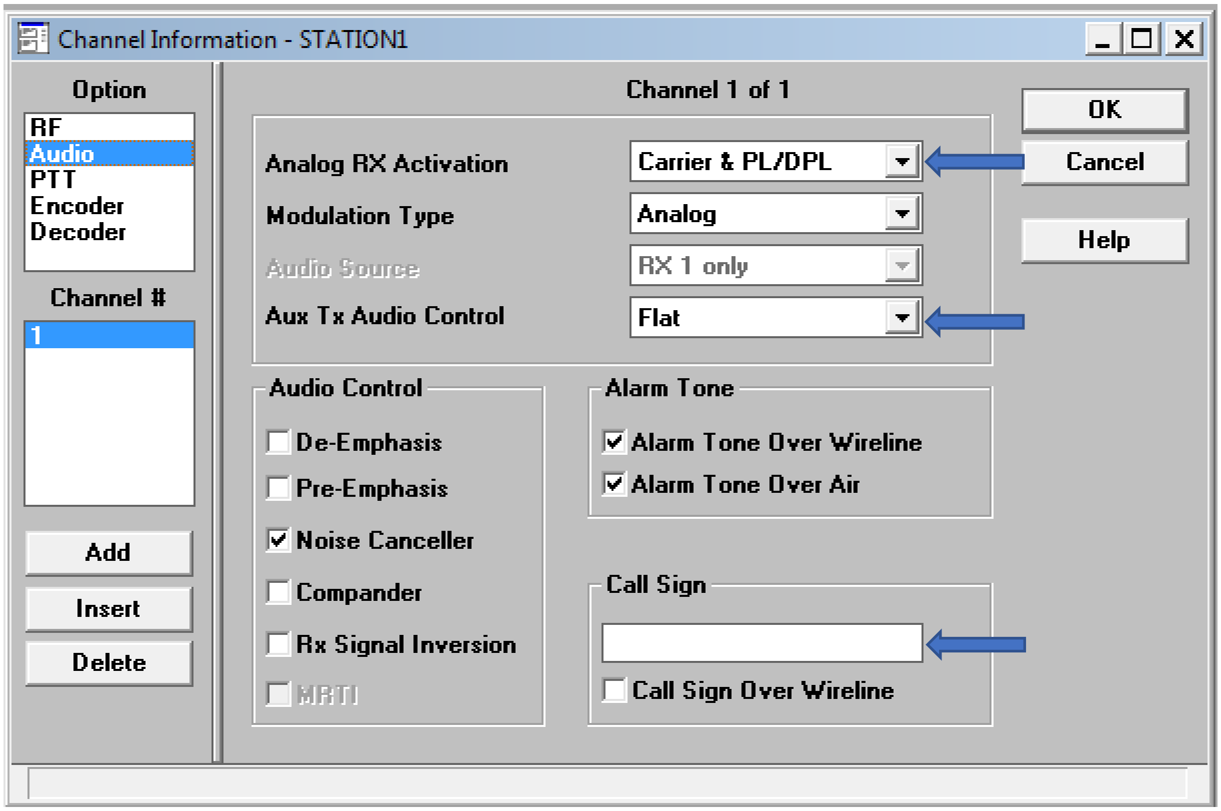
Power Level

Normal 20 Watts

Battery 7 Watts

Uncheck to disable this option

**Audio:**



Channel 1 of 1

Analog RX Activation: Carrier & PL/DPL

Modulation Type: Analog

Audio Source: RX 1 only

Aux Tx Audio Control: Flat

Audio Control

- ☐ De-Emphasis
- ☐ Pre-Emphasis
- ☑ Noise Canceller
- ☐ Compander
- ☐ Rx Signal Inversion
- ☐ MRTI

Alarm Tone

- ☑ Alarm Tone Over Wireline
- ☑ Alarm Tone Over Air

Call Sign

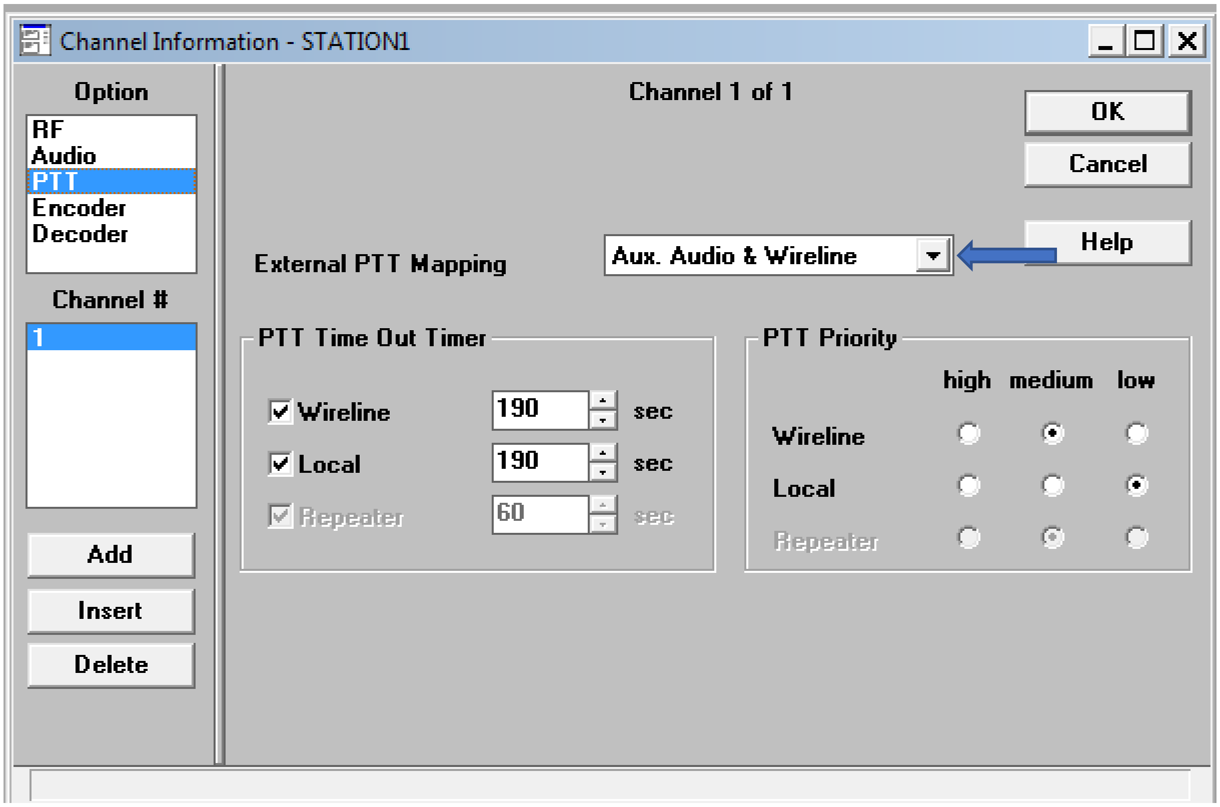
- ☐ Call Sign Over Wireline

- Analog RX Activation = Carrier & PL/DPL
- Aux TX Audio Control = Flat (This should have been set during the STM32 setup)
- Call Sign: If you want to use the repeater ID, then enter your callsign here and check the box for Call Sign Over Wireline. However, note that the MTR2k will attempt to ID at the interval time, but if it is interrupted, it will stop and wait a short while before attempting again. If during the re-attempt it is interrupted again, it will again stop, wait and try again. Thus, if there is a QSO in progress, and the users do not allow time for the repeater to ID, it is possible that the repeater will never ID until there is a long enough pause for the repeater to ID. Essentially, there is a possibility that the repeater may not ID every ten minutes in accordance with FCC regulations.

  Due to the possibility of non-compliance, I initially chose not to use the MTR2k ID function, and set Pi-Star to CW ID every 10 minutes. For analog, I have AllStar take care of the station ID which allows more controls over how the ID is transmitted (voice or CW) as well as additional timing controls. Since the station ID is sent from AllStar, the repeater will encode PL with the transmission. To handle the station ID for the digital transmissions, I configured Pi-Star for CW ID. Like the MTR2k, the CW ID sent from Pi-Star does not encode PL when transmitted. Also note that Pi-Star will send the CW ID every 10 minutes regardless like a beacon, not only when there is traffic. However, if there is analog traffic, the Pi-Star will "lockout" as a result of the Wildcard settings. If Pi-Star is in "Lockout," it will not send the CW ID, which is why AllStar must handle the ID for analog.

  Further testing of the CW ID function in Pi-Star revealed that it also has the same problem as the CW ID from the MTR2k. I.e., if there is traffic (digital or analog), and the users do not pause long enough for the system to ID, it will not send the CW ID. This creates the same problem of possible non-compliance. As such, I ultimately chose to enable the CW ID from the MTR2k instead of Pi-Star, since it will only attempt to ID if there is traffic, and not beacon it every 10 minutes regardless.

**PTT:**


Channel Information - STATION1

Channel 1 of 1

External PTT Mapping: Aux. Audio & Wireline

PTT Time Out Timer

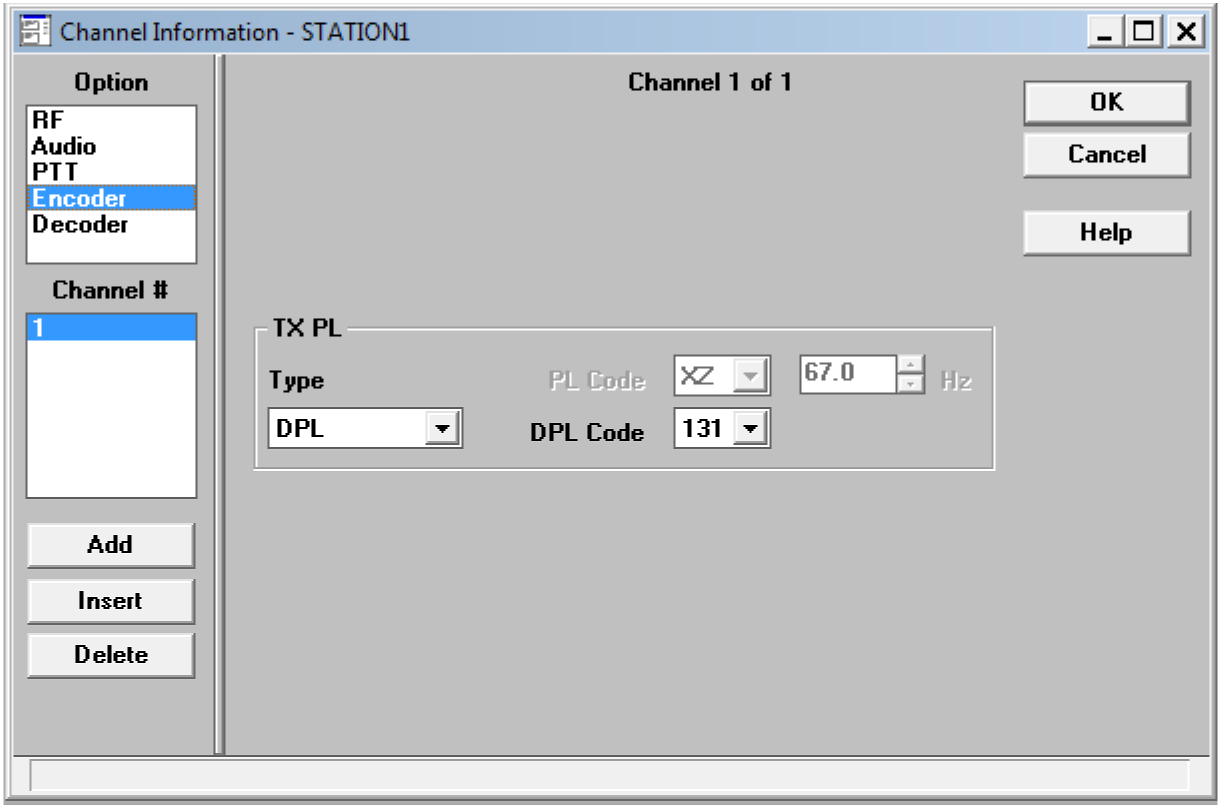
- ☑ Wireline 190 sec
- ☑ Local 190 sec
- ☑ Repeater 60 sec

PTT Priority

| | high | medium | low |
|---|---|---|---|
| Wireline | ○ | ● | ○ |
| Local | ○ | ○ | ● |
| Repeater | ○ | ● | ○ |

You must set External PTT Mapping to “Aux. Audio & Wireline.” The STM32 card uses the Aux. Audio channel, while AllStar will use Wireline. You can also setup your time-out-timers on this screen. I usually set mine to 3 minutes (180 seconds) in AllStar. I added 10 seconds of padding just in case AllStar decides to send the station ID after a transmission.

**Encoder/Decoder:** The next two screens are where you set the PL (CTCSS/DCS) encode and decode settings.


Channel Information - STATION1

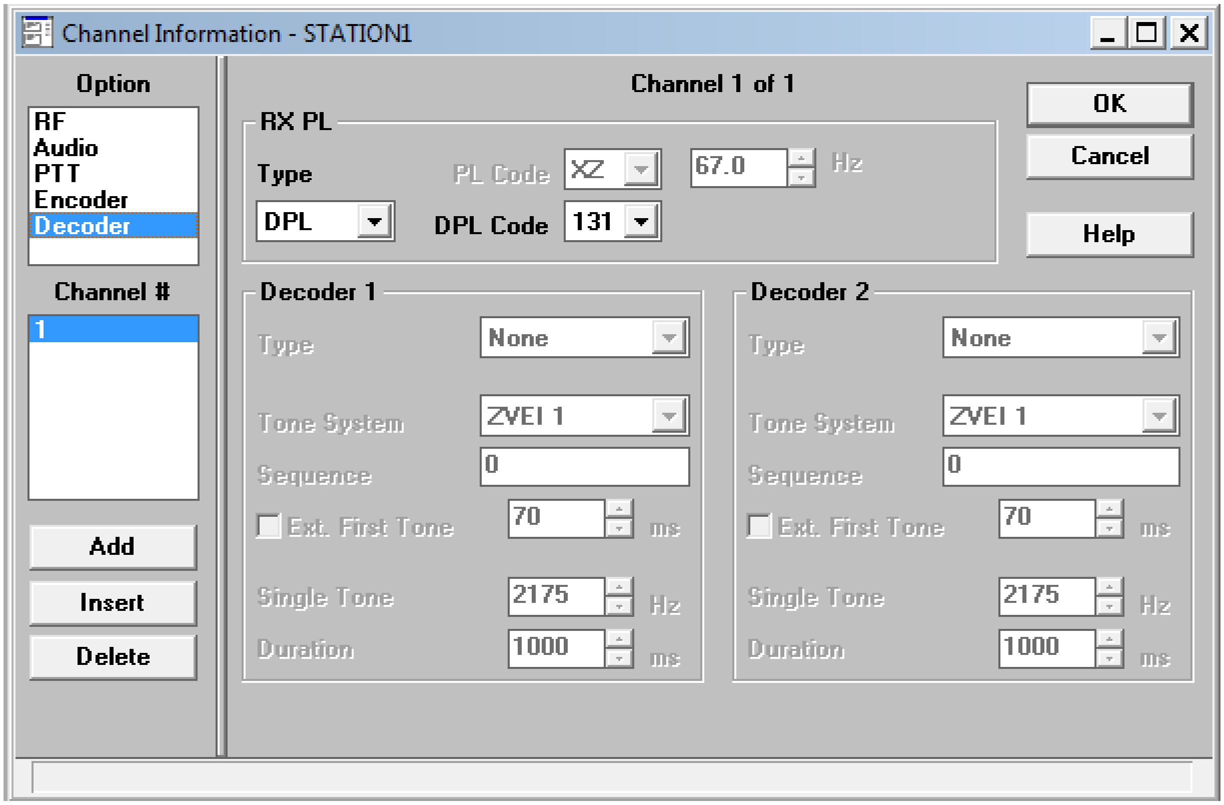
Channel 1 of 1

TX PL

Type: DPL

PL Code: XZ 67.0 Hz

DPL Code: 131



Channel 1 of 1

RX PL

Type: DPL

PL Code: XZ 67.0 Hz

DPL Code: 131

Decoder 1

Type: None

Tone System: ZVEI 1

Sequence: 0

□ Ext. First Tone: 70 ms

Single Tone: 2175 Hz

Duration: 1000 ms

Decoder 2

Type: None

Tone System: ZVEI 1

Sequence: 0

□ Ext. First Tone: 70 ms

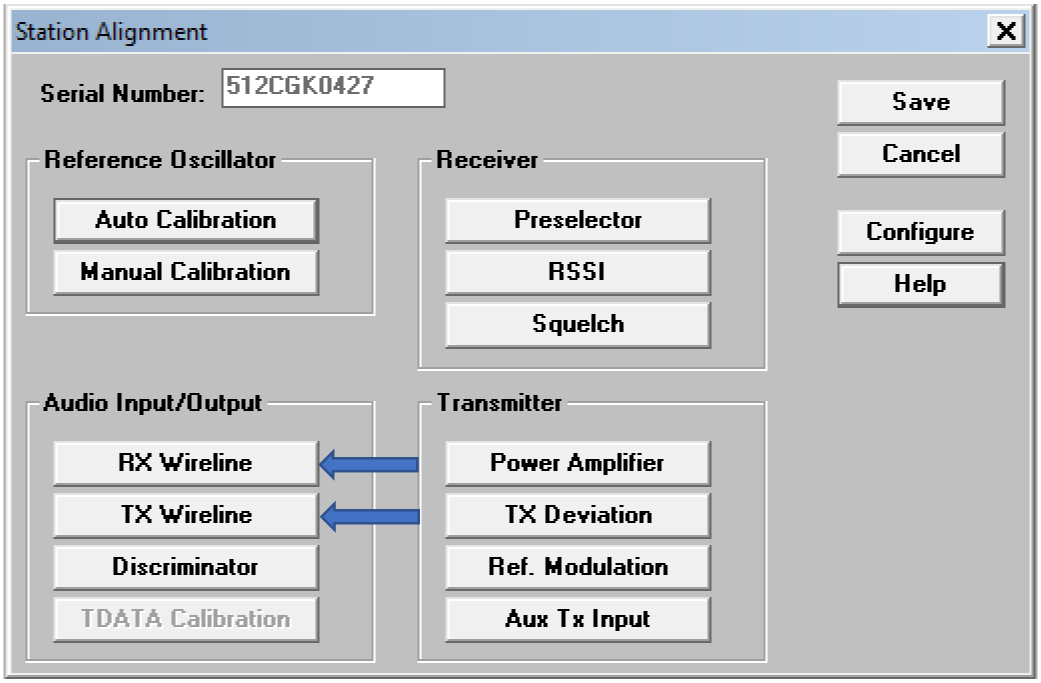
Single Tone: 2175 Hz

Duration: 1000 ms

***__Wireline Audio Level Calibration:__***

I do not want to go into details of how to calibrate the MTR2k, as it is a topic all its own and requires special equipment, but I feel it is necessary to touch on this one point. When setting up my MTR2k, the Wireline audio was not calibrated which resulted in no audio coming out of the repeater. If this happens, try aligning the Wireline TX/RX audio levels.

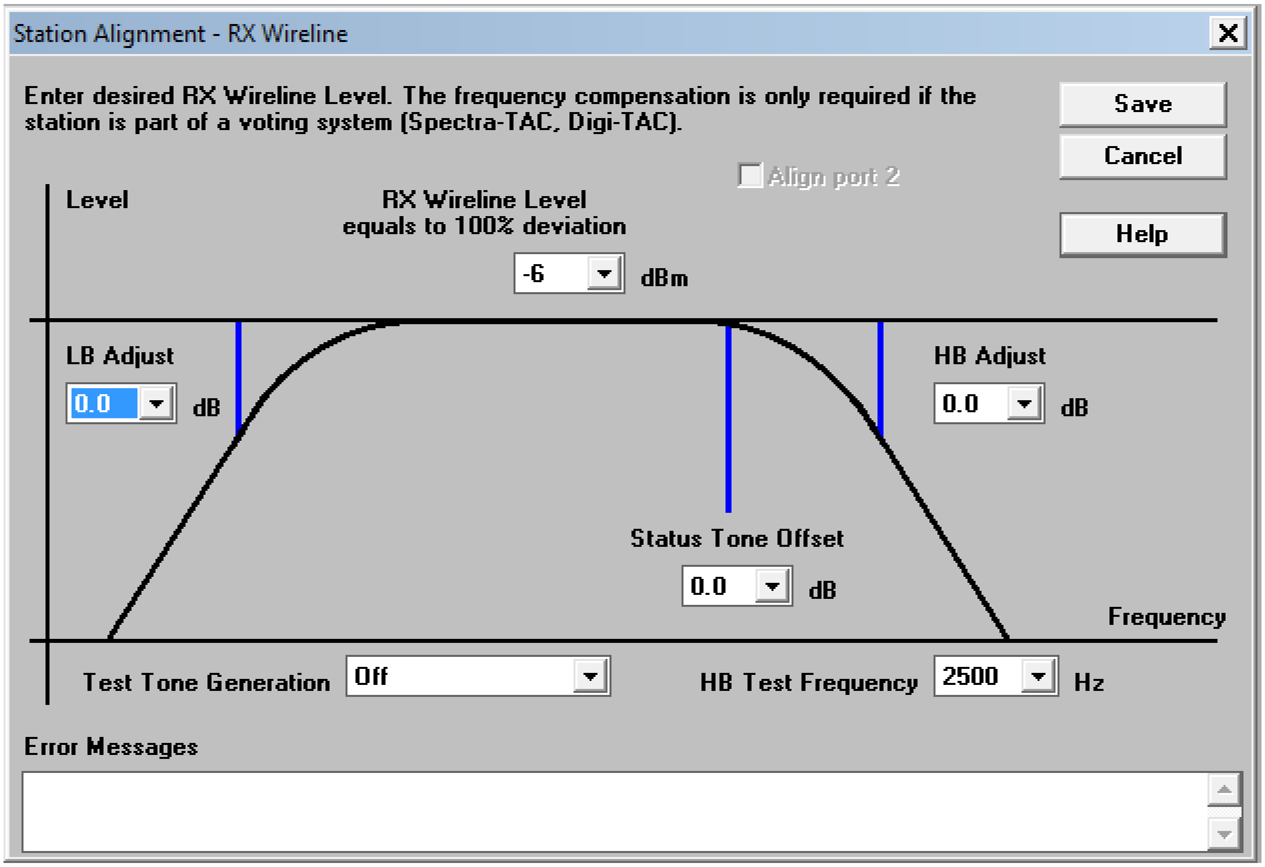
From the Service menu, select Station Alignment. The following window will appear.



Serial Number: 512CGK0427

## *RX Wireline:*

Only the top setting for RX Wireline Level needs to be set.  I set mine to -6dBm.  The other settings are only used for voting systems and therefore can be ignored or set to 0.0.



Station Alignment - RX Wireline

Enter desired RX Wireline Level. The frequency compensation is only required if the station is part of a voting system (Spectra-TAC, Digi-TAC).

RX Wireline Level equals to 100% deviation: -6 dBm

LB Adjust: 0.0 dB

HB Adjust: 0.0 dB

Status Tone Offset: 0.0 dB

Test Tone Generation: Off

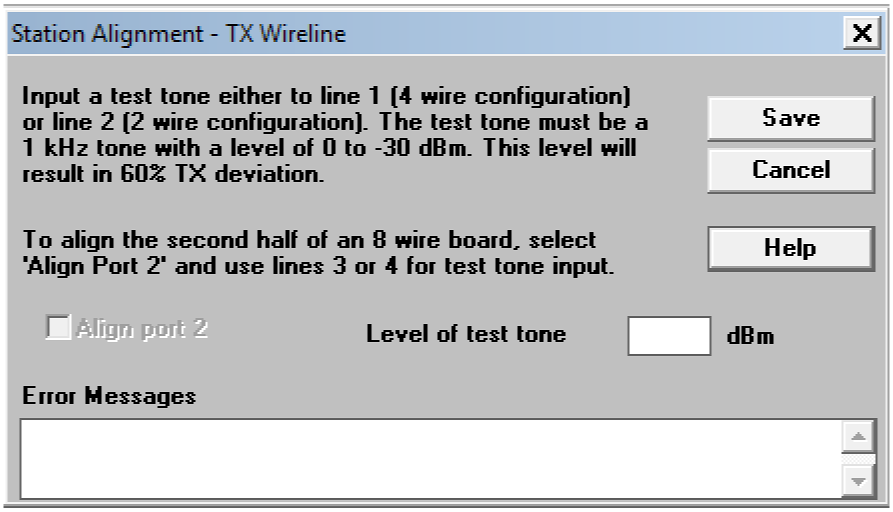
HB Test Frequency: 2500 Hz

Error Messages

## *TX Wireline:*

This is the more important setting, and a bit trickier.  You need to connect a tone generator to the Wireline TX inputs on the rear of the MTR2k (Wireline 1-, 1+).  Set the generator to produce a 1khz tone at a level of your choice between 0 to -30 dBm.  I chose to go with -10dBm.  Once you are injecting the signal at the desired level, you simply enter that level into the field on the screen below and click Save.  The MTR2k will listen to the tone input and calibrate based on the level you entered.  However, I discovered that calibrating mine at -10dBm produced very low audio levels from the MTR2k.

After a lot of research into the low audio level issue, I found some posts in forums related to calibrating the Wireline levels which indicated the MTR2k's are notorious for low audio.  To correct this issue, one post suggested that you inject the 1khz tone at a level lower than what you enter in the MTR2k.  i.e. inject a 1khz tone at -14dBm but enter -10dBm into the field on the calibration screen.  Doing this tricks the MTR2k into increasing the audio level.  I used -13dBm and set the MTR2k at -10dBm, which brought my audio levels up closer to where it should be, then did fine tuning using AllStar.  You may need to try different levels to adjust to your needs.



Station Alignment - TX Wireline

Input a test tone either to line 1 (4 wire configuration) or line 2 (2 wire configuration). The test tone must be a 1 kHz tone with a level of 0 to -30 dBm. This level will result in 60% TX deviation.

To align the second half of an 8 wire board, select 'Align Port 2' and use lines 3 or 4 for test tone input.

***Final Settings:***

The last part that is needed is to configure your AllStar simpleUSB audio settings. Below are the simple USB settings I use. You will need to adjust the audio levels according to your specifications, but this should get it working.

Simple USB Settings:

- Pre-emphasis = Enabled
- De-emphasis = Enabled
- PLfilter Mode = Disabled
- DCSfilter Mode = Disabled
- PTT Mode = Currently active LOW
- COSFROM Mode = usb
- CTCSSFROM Mode = usbinvert

Hopefully this guide will help people who are interested in using the MTR2k repeater as a mixed-mode repeater with AllStar.

Special thanks to the following people for their help:

Patrick Dupont, AD5MT
Ian Stewart, AJ6GZ
Cort Buffington, N0MJS – for building and designing the STM32DVM board, and his how to articles.

Guide written by Larry Aycock, N6LKA.