Battery Hints and the FT-70

Category: Common Features of Radio Types
October 6, 2020
What’s up with the FT-70 battery indicator? Why does it seem like the battery doesn’t last very log?

The FT-70 doesn’t actually know the state of charge of the battery. It only knows the voltage – which can be misleading while the battery is being charged. The Battery Management Unit within the battery pack is responsible for managing the charging and discharging of the battery. While computers and smart phones generally use the BMU to display state of charge, I’ve never seen an HT that does that.

You are correct that the charging indicator does not show a battery level, it displays a timed barograph that slowly fills in since the HT has no data from the BMU.

Also note that the first several charge/discharge cycles on a Li-ion battery will be vastly different from the subsequent cycles. The first couple of cycles are need to condition the battery. Generally capacity will increase for the first few cycles.

To have the battery last as long as possible:
1. Don’t over discharge the battery. My rule-of-thumb is to stop using the battery (if convenient) when the Rx voltage reaches 7.4 V. At this point 80% of the battery capacity has been used. Most of the capacity is given up between 7.4 and 8.0 volts.
2. Don’t charge a hot battery. Heat is the enemy of Li-ion. It really likes nice, comfortable room temperatures around 70F/20C. Best not to leave it in the sun in a parked car!!!
3. Don’t overcharge. It’s fine to pull the battery out of the charger before it is fully charged. Generally the last 10%

(8.0V+) goes slower because of the reduced charge current. Pulling it out at 8.0V will extend the number of charge cycles and have very little impact on stored capacity.
4. Avoid using 5 watts. At high power the radio generates a lot of heat – with the heat sink right up against the battery. It also places the greatest demand on the battery – especially if the battery is more than 70% discharged. The higher discharge current also creates more internal heating within the battery.

What Does the HRI-200 Actually Do?

Category: WiRES-X
October 6, 2020
What's the difference between using an HRI-200 and directly connecting to the radio?

1. It is the only option for connecting a repeater to WiRES-X.
2. It uses UDP communications sent directly to the Room. So network communications are very efficient.
PDN uses TCP instead of UDP. (Why use UDP? UDP is simple. If anything gets lost, it is dropped. Good for real-time communications. If a packet is missed with TCP, it will keep retrying up to a limit which is not good for real-time. VoIP, VPN's, Netflix, etc. all use UDP.)

The PDN TCP packets (non-HRI-200) are sent to a server in Japan. This server converts the packets to UDP and then sends them to the selected room in the same way the HRI-200 does. So by using the HRI-200 you bypass a drip to Japan and a potentially

problematic conversion from TCP to UDP and back to TCP for the return trip to your station.

Yaesu uses the TCP approach to get around forwarding of ports. The WiRES-X PDN software establishes a TCP connection with the Yaesu server. It's through that connection stations can contact you even though you don't have an open port. With the HRI-200, stations just get your IP address and send you packets directly. This is why PDN cannot efficiently host a room – all room traffic would have to go to Japan for the conversion process.

In a previous post (2016) I described what the HRI-200 does. Why it is the way it is is somewhat historical from the history of WiRES, WiRES-II, and now WiRES-X. Obviously an Ethernet port on the back with an embedded processor to run things would be a better way to go. I can only guess that the ROI is simply not there for Yaesu. They do, after all, have to make a profit.

One further note. It's probably a good thing non-Yaesu equipment doesn't connect to WiRES-X. The biggest problem in bridging hotspots (YSF, FCS, etc.) to WiRES-X is that the hotspots are bad neighbors. They generate bad packets, do crazy things, and can really trash a network. As a room operator we have no control over who tries to connect, so at least the Yaesu equipment has gone through compatibility testing. Believe me, very little testing is done on hotspot software.

MNWis (21493) bridges to YSF (21493) via a YSF Reflector where I have extensively modified the code to drop bad packets, control information, and all manner of things that shouldn't see there way into the WiRES-X node. I had to do this because we were constantly having hotspots connect and mess everything up. It got very, very tiring working with the (mostly new) hotspot owners to fix their problems. I mean the conversations were endless and constant! So I just dropped anything that looked bad

in the YSF Reflector software and the problems went away as well as a few hotspots who could no longer connect or talk to people.

System engineering is critical. We need Yaesu to play the roll of system engineering to keep all the Fusion products working with each other. It’s so nice that all the Fusion equipment basically works the same way and does the same thing all of the time!

BTW, you can bring your questions/comments to the MNWis Fusion Technical Net held Monday’s at 7:30 PM Central in the above mentioned locations.

WiRES-X Online/Offline

Category: WiRES-X
October 6, 2020
When a WiRES-X node keeps going online then offline, this type of problem is almost always the result of a networking issue. A networking problem will occur and will wack out the HRI-200 software. Once the software has been wacked out, it will continue to cycle in and out of whatever room it is connected to. Restarting the WiRES-X software usually solves the problem.

Router problems:
1. Turn off UPnP. It should never be used!!!!!!
2. Turn off all quality of service features like those to improve Xbox gaming or VoIP functions. (Yes I know we’re using VoIP so you’d think it would help.)
3. In fact turn off everything you don’t have to have.
4. Some routers are total crap and need to be rebooted every few

weeks.
5. You’re not using Wi-Fi are you? Wired is always better than wireless – we should know! Drop-outs in Wi-Fi can cause this issue.

ISP problems:
1. Something as simple as the ISP dropping your connection for a few minutes in the middle of the night when they’re doing maintenance.
2. The ISP just doing a bad job of getting packets to you during prime evening Netflix viewing.
3. The ISP switching your IP address. (I think CenturyLink just does this for the fun of it.)

Computer:
1. Make sure Win10 doesn’t put the USB interface to the HRI-200 to sleep. It really, really likes to do this. Just going to the obvious place in the control panel to turn this feature off doesn’t really turn it off. See the documents on running Win 7/10 24x7x365 remotely in the Fusion Help section at HamOperator.com.
2. Make sure the computer isn’t crap. Cheap computers may not have quite the reliable communication interfaces that we need.
3. Banish all RFI and ground loops. Make sure RF is not getting into the USB interface to the HRI-200. This can cause message errors and really screw things up.
4. Make sure you computer actually has enough power to run the HRI-200 reliably. Get one of those USB power thingies and make sure you have a solid 5VDC under load.

Over the air:
1. Bad data on the WiRES-X network can mess things up. This happens mostly with hotspots that are bridged into the WiRES-X room. On MNWis and a few other networks we’ve banished this problem by banishing FCS and running YSF server software that

banishes bad data from hotspots. So if you have this problem in one room, say AmericaLink and not others, there may be nothing you can do.
2. Yaesu radios that are running really old firmware or have not had the firmware updated correctly (I.e., “I did the main CPU but I’ll get around to the DSP later.”)

Side note:
The port check should not be relied upon to definitely prove it’s working or not working. The test is not exactly the same as actually communicating with the Yaesu list servers and the room server. So it can lie to you with false positives and false negatives.

You may also experience a similar problem if the Yaesu list servers are not able to keep up with the demand or if maintenance is being done. (When they do maintenance in the wee hours of the morning, that’s right in the middle of the day for us.)

And keep in mind the MNWis Fusion Technical Net held in MNWis WiRES-X and YSF # 21,493 Monday nights at 7:30 PM Central.

WiRES-X and USB Problems

Category: WiRES-X
October 6, 2020
If you’re having problems with USB problems when using WiRES-X with or without other programs, read on.

The WiRES-X software is pretty good at identifying the HRI-200 vs. connecting directly to the radio. Windows, however, is very

bad at managing USB ports. The WiRES-X software makes two USB connections: One as a virtual COM port and the other as a sound device. Windows is also very bad at managing sound devices.

Virtual COM ports can be identified by the port number (a real pain as it is always changing – see below) or by identifying the hardware via the manufacturer and product number that is burned into the USB device. Generally software that does not provide an option to specify the COM port (as with the HRI-200) is using the hardware identification to establish the association.

When Windows sees a USB device it will enumerate it and create a driver for it. If you move it to another port it will, for reasons we can only guess at, create another driver. It will keep making new drivers as long as you have USB ports and USB devices left. Why, we can only guess. Periodically it may be necessary to go through a clean out all of these "drivers" that Windows keeps making and saving until the end of time. You can do this from the command line or from within Device Manager.

I have found WiRES-X to play nicely with other devices. For example I have no problem running hotspots, Ham Operator Deluxe, WSJT-X, etc. on the same computer that is running WiRES-X.

You may find it beneficial to use a USB 2.0 vs. 3.0 port. Why? USB limits the number of connections that can made per USB controller. Keep in mind that several of the ports on you computer may use the same controller. Also, each USB device may use more than one connection. As I recall sound devices generally use two connections: One for the audio and one for control. A USB 2.0 controller supports 255 connections. How do I know? I actually ran up against this limit using a couple of 16 port USB hubs. USB 3.0, while faster, does not permit as many connections. I don't remember the exact number as it has been some time, but the number SIX sticks in my brain. As you might

imagine, it is very easy to burn up six connections. Also keep in mind that the USB controller may be servicing hardware that is internal to the computer. It all depends on how the computer was designed. Top end computers will generally have a higher controller to port ratio. Bottom line: you may want to use USB 2.0 if you have been using USB 3.0.

Perhaps try something like this:
1. Blow away the WiRES-X and any conflicting software. Make sure everything is unplugged from USB.
2. Blow away all of the USB drivers. Just uninstall them. Don’t worry, the files to recompile a driver are still on the computer. [Device Manager with admin. Enable option to view all devices. Delete drivers under COM ports. Delete drivers under sound devices (except for hardware built into your computer.)]
3. Install the WiRES-X software. Allow it to associate with the USB devices it needs by plugging in the HRI-200, etc. If you’re not using an HRI-200, you may find it beneficial to use an external USB audio device for WiRES-X. If you can use USB 2.0 ports.
4. With WiRES-X happy running, install the other software. Plug in the necessary USB devices so it can similarly make the association. At this point everybody should be happy with both programs running just fine.
5. Write down which device was plugged into which USB port. In the future, always use the same USB port for the devices. We don’t want Windows creating more drivers.

Good luck solving the USB problems!

Crimp Vs. Solder

Category: Projects
October 6, 2020
What’s best, crimping or soldering. Should you use crimping? How is a good crimp done.

Below are references that address these questions.

Wikipedia Entry

Crimping vs Soldering Advantages and Disadvantages

Good and Bad RF Crimp Connections

ARRL Instructions for Crimping RF Connectors

How to Make a Quality Crimp

More on Making a Quality Crimp (technical)

Molex Crimping Manual (technical)

TE Systems Crimp Chart

Notes on crimping:

It is always best to have the right tools. Make sure you use the right connector for the cable and the right crimp die for the connector. With the right tools, RF connectors can be crimped very quickly and are very reliable. One nice advantage, is that it is easy to reuse the connector. The center pin is generally soldered (which is easy to unsolder) and the shield is crimped with a short metal “tube”. Some companies sell bags of replacement “tubes” so the connectors can be reused. Follow the connector’s instructions EXACTLY and you’ll have a solid connection that will last forever.

If you are making outdoor connections, I recommend using N connectors as they are hermetically sealed. (PL-259s are not) For power connections, I recommend using the same connectors that are used to connect solar panels together. They're much cheaper than Anderson Power Poles and they form a hermetic seal. For signal connectors, go to your hardware store and look in the irrigation section. They have special twist-on connectors that are meant to be underground or sit in the rain. They're pretty cheap if you buy a big bag.

Check back again in about a week and I'll have more links and references.

MMDVM .96″ OLED Now Works!

Category: Hotspots
October 6, 2020
It has really been bugging me. I've never been able to get a 0.96″ OLED to work on my Pi-star hotspots. The 1.3″ works okay. Why not the 0.96″. Here's why:

The 1.3″ screen uses the SSD1306 controller. Adafruit uses this controller on all their OLED screens. The (cheap) Chineese OLED screens use an older SSD1106 controller.

Raspberry Pi uses the Adafruit OLED driver software, so Pi-star does also. Naturally Adafruit has no desire to support the SSD1106 as they don't sell anything that uses it. In other words, the standard Pi-star distibution OLED driver doesn't work with the SSD1106 .96″ displays.

What's different? The SSD1106 has a slightly smaller refresh RAM

array than the SSD1306. That causes each line of pixels to be off by 2 hence small valid text ends up looking like garbage. The graphics still look okay.

The solution is to replace the Adafruit driver with a modified version that supports the SSD1106. The driver is located at /usr/local/lib/ArduiPi_OLED.so.1. I have built a replacement driver with the SSD1106 support.

1. SSH into your Pi-Star (can be done from the Expert screen).
2. Switch to root with “sudo su”.
3. Make the disk RW with the command “rpi-rw”
4. Move to the directory where the driver is installed with “cd /usr/local/lib”.
5. Copy the driver file from HamOperator with the command: “wget http://HamOperator.com/files/libArduiPi_OLED.so.1.0”.
6. Change the permissions on the driver with the command: “chmod 777 /usr/local/lib/ArduiPi_OLED.so.1.0”.
   (I don’t know why ‘777’ ‘755’ should work but the original was ‘777’.
7. Reboot
8. The OLED type needs to be ‘3’ for the 0.96 display and ‘6’ for the 1.3″ display (set in Expert->MMDVM Host). The OLED must also be enabled from the configuration page

Thanks to Charles for doing the driver work!

If you want to download the file directly, you can do that here: libArduiPi_OLED.so.1.0

YSF Reflector Software

Category: Fusion
October 6, 2020
The HamOperator YSF Reflector is currently supporting MNWis, US Kentucky, and America-Link. A test reflector runs at US K9EQ.

A project has been going on at the HamOperator to improve the YSF Reflector software. Why do this?

The original software is very basic and offers very little capability and control. It also creates a HUGE problem when trying to bridge YSF hotspots to WiRES-X rooms because it just passes the data along with no filtering. That means whatever garbage gets sent to the reflector, it doesn't even need to be Fusion, that same data is sent to every connection including the bridged WiRES-X node. When the WiRES-X node gets this data it can misbehave or crash. Hence a project to fix these problems.

And while we're fixing problems, why not make it better by giving it more capabilities?

Current differences between the standard YSF Reflector software and the K9EQ version:

- Decodes all meta data including FICH and all data fields for every mode.
- Extensive logging allows selected logging of parameters needed to determine "what's going on?".
- Filters to drop packets if:
    - The FICH is invalid
    - The FICH does not pass sanity checks
    - Data Wide packets

        - Wires-X control packets
    - Outputs a text file of connected nodes to simplify dashboards.
    - Performance reporting
    - Options can be changed in a *.ini file
    - Ability to host multiple YSF Reflectors on a single server
    - Internal documentation of the program (comments)
    - Several builds are available:
        - Windows
        - Linux
        - Debian server (MNWis and US Kentucky are running in the “cloud”)

Future enhancements will include:

- Drop problematic WiRES-X packets with invalid room/node number in the CSD1 field
    - Future improvement: Edit the packets to remove the problematic information
- Ability to black list callsigns and IP addresses.
- Kerchunk filter to drop short key-ups
- Remote control and status reporting of the reflector via Fusion text messages
- Integrated dashboard removes need for php and greatly reduces CPU overhead and network bandwidth
- Programmed reporting of audio levels informing users if their mic level is too high or too low
- Integration with IMRS
- New, engineered and reviewed API to more tightly integrate hotspots with the reflector
- Ability to programmatically send messages
- Ability to programmatically edit data fields on the fly (i.e., change GPS data, call sign, etc.)
- Improve internal program documentation
- Complete program re-write in Python. This will provide the

ultimate in portability

The reflector will not support non-Fusion communications such as DMR or D*. The reason for this is that non-Fusion systems do not include all of the data that Fusion provides. This would force the reflector to become the lowest common denominator resulting in technology and feature restrictions.

The intent is for our version of YSF Reflector to be open sourced. It will be developed and enhanced with feedback from the Fusion community. By using good engineering practices, our hope is to provide a high quality service that equals or exceeds the existing quality of the Fusion/WiRES-X system. This is an enhancement to WiRES-X, not a replacement.

YSF WiRES-X Unintended Node Switching

Category: Hotspots,WiRES-X
October 6, 2020
A problem exits with YSF/MMDVM hotspots bridging to WiRES-X rooms. Here's what happens.

A user has a hotspot and accesses it using a Fusion HT. They use the WiRES-X control mode to change the hotspot to different YSF Reflectors. Let's say the user wants to access MNWis (Room #21,493):

- They enter the number of the YSF Reflector, "US MNWis 21493", which is 37,624. The hotspot switches to US MNWis 21493 which is bridged to WiRES-X Room # 21,493.

- The user leaves their radio in the WiRES-X control mode. They hear stations, but when they talk, nobody answers and nobody is talking.

Here’s what happened: When they keyed up to transmit, their radio sent a command to switch to Room #37,624. The Wires-X node that was previously connected to #21,493 now connects to #37,624 which belongs to a very nice gentleman in France.

How does this happen?

The Fusion transmitted data includes callsign fields CSD1, CSD2, and others. We’re interested in CSD1. This 10-byte field contains two items: The room number of the connection and; The TxID or DP-ID of the radio. A typical CSD1 looks like 21493F0yxh where 21493 is the room number and F0yxh is the DP-ID. The ‘21493’ in this field will cause a Wires-X node to switch to that room.

When the radio IS NOT in Wires-X control mode, the CSD1 field will be: ‘*****F0yxh’ and no switch will be made.

Bottom line:

DON’T USE WiRES-X control mode to talk through a hotspot or MMDVM!

Workarounds and fixes: (Updated 23-Aug-2019)

Node operators who are experiencing this problem should use the WiRES-X block feature to prevent the node from switching to the room that has the YSF reflector number.

For example: Assume that our node connects to MNWis, Room # 21,493. Our node also provides bridging to “YSF MNWis 21493” which is # 37,624.

- In the View->Node-Info(N) window press “Add”.

- In the “Input ID” dialog box enter the YSF reflector number. In our example this is “37624”.
- Press “OK” then “Close”.
- When a station using the YSF bridge still has WiRES-X control mode enabled accesses the node, the command to switch rooms will be rejected by the WiRES-X software. The software will indicate a rejected attempt to switch rooms.

As an alternative, set the node so that connection changes are not allowed.

- File->Settings->Call Settings->Uncheck “Round Room Connection”.
- Un check “Accept calls while in Round Room QSO.
- Check “Return to Room”
- Fill in the WiRES-X room number. Example: 21493.

YSF server side solutions:

I maintain a version of YSF Reflector that has filters to prevent hotspots causing problems with WiRES-X nodes. My first step is to drop all packets that contain an incorrect WiREX-X room number. Eventually I plan to replace the first 5 bytes of CSD1 with ‘*****’. This last step is difficult to do because it requires the reflector to decode the data, modify it, then recompute the CRC. This also involves working with the interleaving and forward error correction. It is obviously easier to just drop the packets, but that may confuse people on the YSF side since nobody on the WiRES-X side will hear them.

If you are interested in running the enhanced YSF Reflector software that will fix this problem at some point, please contact me privately.

73,

Chris, K9EQ

Old Stuff – Computers

Category: Old Stuff
October 6, 2020
How can computers be “old stuff”? We’ll see…..

The Home First Computers

This week we take a look at three pioneers in the computer business:

- From 1975 MITS’ catalog featuring the Altair 8800 computer. (0.6 MB)
- From the 1977 Southwest Technical Products catalog with their 6800-based computer. (5 MB)
- From 1978 The Commodore PET catalog (0.5 MB)
- From 1979 The Texas Instruments TI-99 (1.2 MB)
- And from 1980 Ohio Scientific catalog with their products. (7 MB)

  How far we have come! Be sure to check out memory size and prices!

Old Stuff – Reference

Category: Old Stuff
October 6, 2020

Techniques and Projects

Here is a variety of new old stuff that focuses on electronics and radio techniques and projects.

- 1958 From Popular Electronics magazine, The Experimenter’s Handbook (15 MB)
- 1965 Again from Popular Electronics, The Experimenter’s Handbook (8 MB)
- 1973 Electronics Theory Handbook (4 MB)
- 1973 Understanding Electronic Circuits (7 MB)

Handbook, Dictionary, and FixIt

For the New Old Stuff for this week, we present some reference materials.

- The 1936 ARRL Handbook 14th Edition (54 MB)
- The 1944 Allied Dictionary of Radio Terms (4 MB)
- From Popular Science in 1946, the Third Radio Make It, Fix It Annual (6 MB)

Tubes, Tubes, tubes….

- From 1951, a manual from Eimac, The Care and Feeding of Power Tetrodes (3 MB) You are feeding your tetrodes, aren’t you?
- 1960 offers a document on Getting the Most out of Vacuum Tubes (13 MB)
- From 1968, the RCA Tube Manual (28 MB)