SOME NOTES ON ROM VERSION 1.2 and ROM VERSION 1.3 - Ed K1MMI
Updated 4/10/2014

A couple of members on this group said the DXLAB suite works with ROM 1.2 so I installed the DXLAB suite and posted some questions(got answers too) on the DXLAB Yahoo Group and I think I have finally figured out the difference between Microcontroller/ROM 1.2 and ROM 1.3.

I downloaded a copy of the FT-990 Operating Manual and it re-defines the UPDATE command to have five options. The FT-990 manual that came with my FT-990 ROM 1.3 defines the UPDATE command to always return 1492 bytes which is how the UPDATE works for ROM 1.2.

The 990 has a Microcontroller (M37700...) and it contains an internal ROM area.
I believe Yaesu supplied the manufacturer with a datafile that contains the ROM firmware. When Yaesu ordered Microcontroller chips for the 990 the manufacturer made the chip with the 990 ROM firmware. Whereas with the FT-1000 there is a separate EPROM which can be reprogrammed like a 1000 times so in 2014 it is easy to get the latest FT-1000 EPROM version. Be it by duping it on a EPROM machine or buying the programmed EPROM via sellers on ebay. With the 990 the Microcontroller is obsolete and if someone can find a Microcontroller with ROM 1.3 it was probably manufactured a good 10 years ago.

I believe ROM 1.2 was around for about a year. My guess would be April 1991 to April 1992.

NOTE: When Yaesu came out with ROM 1.3 in about May 1992 they were offering free upgrades from ROM 1.2 to 1.3. Owners had to send in the Control Unit and Yaesu would replace the Microcontroller that contains an internal ROM. Don't know how long Yaesu offered the free upgrade but am pretty certain it was still being done into 1993. The free upgrade wasn't publicized by Yaesu so most 990 owners who had ROM 1.2 never knew about it. Plus most 990 owners who had ROM 1.2 either didn't have a Computer or a Program that used the CAT port - so it was not a problem.

For ROM 1.2 the UPDATE command always returns the complete status info that includes the two VFOs and all 90 memories resulting in 1492 bytes and a transmission time of 2.984 seconds at 4800 baud.

For ROM 1.3 the UPDATE command was modified to have 5 options. These options gave the Application Program a way to read either or both VFOs, or a single memory channel, or the complete status info that was increased to 1508 bytes. The option that returned both VFO frequencies resulted in 32 bytes being returned in 64 milliseonds.

Usually, when an Application Program on Windows runs it wants to poll the FT-990 several times a second so when the operator moves the Main Tuning dial it can immediately update the frequency display and respond in realtime.

When a program like HRD (Ham Radio Deluxe) sends several special UPDATE commands every second to a ROM 1.2, it returns 1492 bytes for each UPDATE, which takes like 3 seconds for each UPDATE and things get hosed. When HRD sends the UPDATE command it also sets a data byte requesting 32 bytes but ROM 1.2 always returns 1492 bytes.

When a program like HRD sends several special UPDATE commands every second to a ROM 1.3 it returns 32 bytes, making it easy for HRD to display the current 990 frequency in realtime.

With the DXLAB suite there are two choices for a FT-990. One choice is for ROM 1.2. In that case it sends the UPDATE command only one time when the UPDATE button in the COMMANDER window is clicked on. This results in a 1492 byte reply from the 990 and it gets the VFO and mode info from the 1492 bytes.

For ROM 1.3 the special UPDATE option that returns the two VFO frequencies is used. It keeps sending several UPDATE commands every second but it only takes 64 milliseconds to get the info and COMMANDER can update the window to display the 990 vfo frequencies in realtime.

This explains, to me, why the DXLAB suite works with ROM 1.2 and gives owners of a 990 with ROM 1.2

the opportunity to make contacts on the Digital Modes. It also explains why HRD fails when connected to a 990 with ROM 1.2. (I suspect if HRD developers were motivated they could add support for a 2nd FT-990 that contains ROM 1.2. The same as DXLAB developers did.)

General Comment: The 990 like most obsolete radios have many unique parts and it can be a problem obtaining a unique part when the radio manufacturer no longer repairs the radio or doesn't have the obsolete unique part in stock. The source for obsolete parts(ROM 1.3) is from broken 990 radios or when Hams intentionally disassemble a 990 and sell off the individual parts via Ebay or some other website. Fortunately, most 990 failures involve common everyday components, be it resistors, capacitors, diodes, or transistors and most 20+ year old 990's seem to be still running.

SOME ADDITIONAL NOTES:

The Microcontroller/ROM 1.3 chip is on the Control Unit and also contains an internal RAM area. The RAM area is used to save all Memory Channel, VFO Info, and the state of the 990 when it is powered down. When the 990 is powered down the Battery on the Control Unit applies a low voltage to the Microcontroller so the RAM area data is not lost between power cycles. This way when the 990 is turned back on the 990 will be in the same state as the last time it was powered off.

The Antenna Tuner has a different but similar type of Microcontroller on the Tuner Control Unit, it also contains an internal ROM area that stores Tuner Firmware. It also has an internal RAM area that stores the 39 antenna tuner settings. When the 990 is powered down a battery on the Tuner Control Unit applies a low voltage to the Microcontroller so the RAM area tuner data is not lost between power cycles. This way when the 990 is turned back on the 990 Tuner will remember the last 39 saved antenna tuner settings.

NOTE: The Tuner Metal Case has a hole in it that allows the user to move a switch that disconnects the battery. With the battery is disconnected the Tuner is 100% operational but all Tuner Settings are not saved during power cycles because the backup battery isn't powering the Microcontroller.

M37700M4xxxFP Info:

The M37700M4XXXFP is an 80 pin surface mount Microcontroller. It contains a CPU, Internal ROM that contains the FT-990 Control Program, Internal RAM to keep track of the state of the FT-990 during operation and between power cycles, a UART, A-D converter, and Programmable Ports to define input and output pins. This chip is the heart and soul of how information is communicated between the ROM Control Program and interacting with the operator. The microcontroller contains interfacing hardware that communicates with all the other microprocessors in the FT-990.

Yaesu PN G1091180, M37700M4-117FP contains Rom Version 1.2
Yaesu PN G1091594, M37700M4A215FP contains Rom Version 1.3
Yaesu PN G1091595, M37700E4AFP MY-033 contains Rom Version 1.3 (Chip has a white stick on label with "MY-033" on it. This 990 built June 1992)

NOTE: Yaesu in the USA ran out of ROM Version 1.3 Microcontroller Chips about 10 years ago and Yaesu in the UK ran out about 18 months ago.

CAT PORT:

When I built a homebrew MAX-232 serial interface the three CAT Commands (Update, Read Meter, and Read Flags) all failed. By trial and error I discovered by adding a 1.3K resistor from the Serial Out (Pin 2 on the CAT port) to ground (Pin 1 on the CAT Port) all three CAT Commands worked fine. I chose 1.3K because it was 100% reliable when reading the status data from the Update Command. With the 1.3K resistor my 990 has worked perfect for over 20 years. I am aware that others discovered the same thing and they chose other resistor values - like 1.2K to 2.0K. I have read Yaesu later modified the 990 and

added an internal resistor but don't when that happened.