ICOM UX Module Design Guide

By Joseph Haas, KEØFF
7/27/2025

joeh-at-ke0ff-dot-org

The ICOM UX module series consists of at least 9 different modules covering 7 frequency ranges. Most of the modules were designed for the IC-900 Multi-band Transceiver, while some were designed specifically for the IC-901 Transceiver. The modules feature a simple daisy-chain control bus architecture which allows them to be assembled (also referred to as "stacked") to cover several frequency bands and modulation modes.

This document details the communications formats and interface issues relating to the ICOM UX series modules that were designed to connect with the IC-900 or IC-901 transceivers. It also covers the control of the IC-901 base modules, which requires an understanding of the IC-901 controller signaling frames. It represents a collection of technical data needed to control the ICOM UX modules and is the culmination of many years of effort spent reverse engineering the modules behavior and refining the resulting interface software and hardware. The focus here is to be an "interface control document" that describes the module interfaces to a degree sufficient to allow one to design a remote controller from scratch (no small feat, as I can attest). All of the information contained herein has either been collected by monitoring module signals that are easily accessible, or by analyzing published schematics and component datasheets.

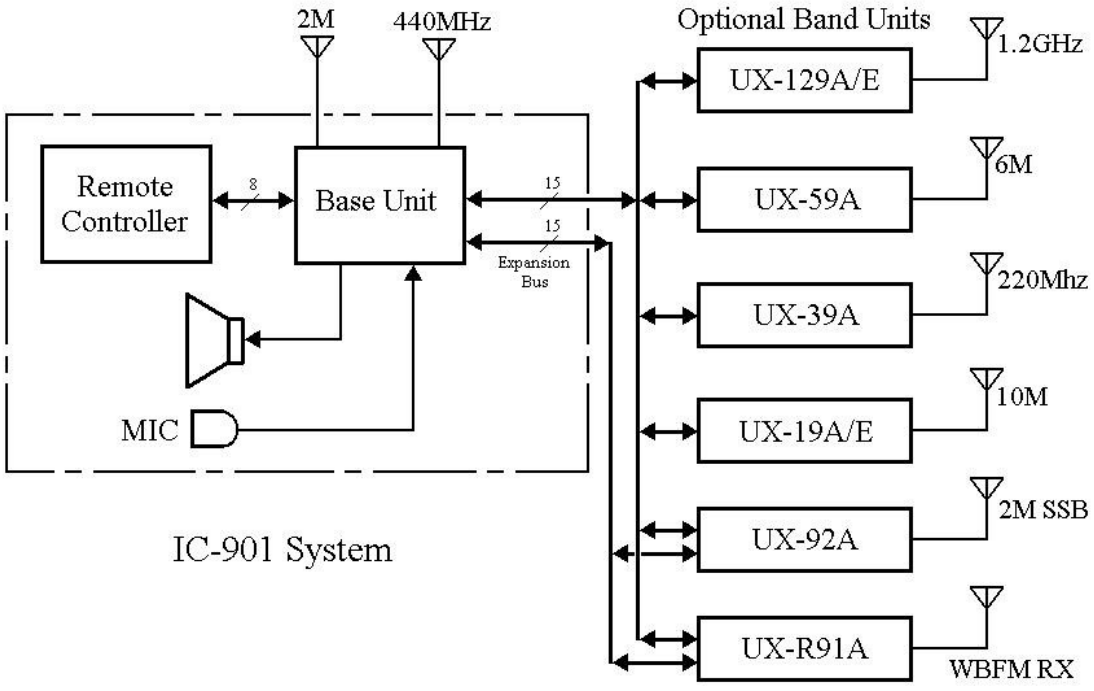


Figure 1a. Simplified block diagram for the IC-901.

© Joseph M. Haas, 2025, all rights reserved

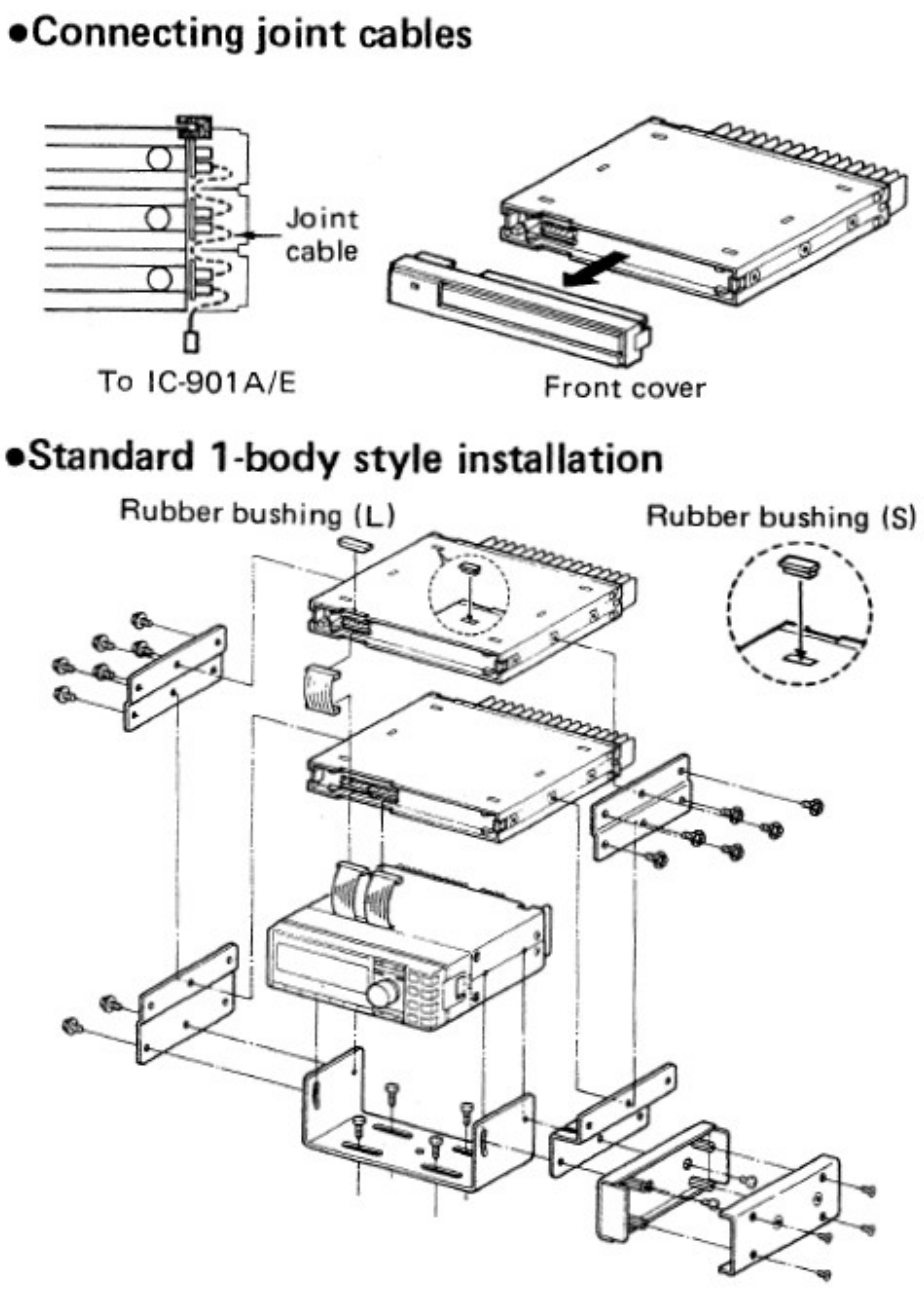


Figure 1b. Exploded assembly view of the IC-901.

Figures 1a and 1b illustrate the IC-901 Transceiver architecture. While functionally similar, there are differences between the IC-901 and IC-900 – the IC-900 block diagram (not shown) is similar except that the 2M and 440MHz transceivers are not built into the base unit, and there is no expansion bus (thus, no expansion modules). The interface to the UX modules is accomplished using a 15-pin daisy-chain connector (two of them, in the case of the expansion modules) which conveys digital and analog signals as a series of short-stub cables (Joint Cables). 13.8Vdc Power and RF connections are at the back side of the modules, and the daisy-chain buses are at the front of the modules.

With a microprocessor and some relatively simple interface circuitry, one can design a control unit that eliminates the need for the base unit and remote controller. This has been done by several companies to

© Joseph M. Haas, 2025, all rights reserved

produce frequency agile, multi-band remote base interfaces for repeaters. I have personally created a very efficient APRS radio as well as frequency agile repeater-based remote-base systems using the FC-900/UX modules and the IC-901/UX modules and have recently integrated this functionality into a replacement front board, allowing the UX module to act as a stand-alone transceiver. Their modest size and lack of display (a source of increased current draw for many radios) make them very attractive for a variety of applications.

The ICOM UX Module

Figure 2 illustrates a typical UX module with the covers removed. The ICOM modules are generally divided into two circuit cards, identified by ICOM as the Main Unit and the Front Unit. The Front Unit is a small circuit card located at the front of the module. It performs the base unit bus-interface and module address selection functions via a pair of 15 pin, daisy-chain "joint" cables (the "expansion" modules have two pair of these connectors). The Front Unit is essentially identical for all of the original IC-900 series modules (differing only in that there are a series of solder-jumpers used to program the module address ID). The Front Units for the IC-901 "expansion" modules are not interchangeable but perform the same functions. The Main Unit holds all of the transmitter and receiver circuits. Some of the Main Unit circuit cards are similar from one module to another (the 10M and 6M modules are an example), but even those are not readily swap-able from one band to another because of component differences.

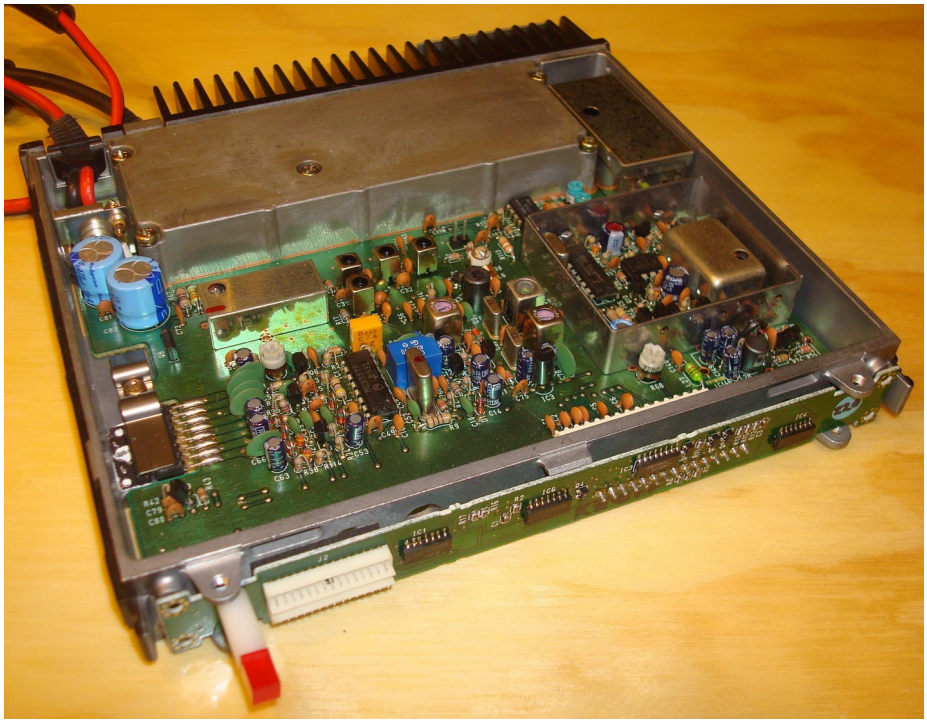


Figure 2. An ICOM UX module with covers removed.

Of particular interest for any given module is the PLL circuit as this is the primary means for controlling the transmit and receive frequencies. The following table describes the modules that are discussed in this document:

© Joseph M. Haas, 2025, all rights reserved

UX Module	band/mode	RF Out (hi/lo)	Compatibility	#daisy-chains
UX-19	10m (FM)	10W/1W	IC-900/901	1
UX-59	6m (FM)	10W/1W	IC-900/901	1
UX-29A	2m (FM)	25W/5W	IC-900	1
UX-29H	2m (FM)	45W/5W	IC-900	1
UX-39	220 (FM)	25W/5W	IC-900/901	1
UX-49	440 (FM)	25W/5W	IC-900	1
UX-129	1.2GHz (FM)	10W/1W	IC-900/901	1
UX-S92A	2m (SSB)	25W/5W (PEP)	IC-901	2
UX-R91	Wide-band RX (AM/FM)	N/A	IC-901	2
IC901-VHF	2m (FM, base unit)	50W/5W	IC-901	n/a
IC901-UHF	440 (FM, base unit)	35W/5W	IC-901	n/a

Based on the schematics for the IC-901, provision was made for several other expansion modules, but I have never seen anything available, nor heard any rumor of any other modules. The Front Units on the expansion modules are not interchangeable as they feature circuitry and connectors that are dedicated to the respective module.

UX Module Interface Specification

This section describes the connectors, pinouts, and circuits needed to interface with the UX modules. A daisy-chain connector system connects each module to the base unit. These connectors (one for the "controller side" and one for the "module side") are located at the front of the module. The daisy-chain connectors (designated as OPC-179 by ICOM, and sometimes referred to as "joint" or "ribbon" cables) are 15 pin single row connectors with a 1.5mm contact pitch. The Front Unit section covers the construction of these cables in more detail. Since the OPC-179 is symmetrical, either end may be used interchangeably.

Primary daisy-chain bus (J1 is lower, J2 is upper):

Pin#		Signal Name	Type	Description
J1-1	(J2-15)	E (GND)	Power	Signal common
J1-2	(J2-14)	+5V	Power (in)	+5V to operate the Front Unit circuits
J1-3	(J2-13)	MOD	Analog (in)	Modulation signal, pre-emphasized, limited, 125mVrms
J1-4	(J2-12)	STB	Digital (in)	Serial strobe signal (+5V CMOS logic levels)
J1-5	(J2-11)	DATA	Digital (in)	Serial data signal (+5V CMOS logic levels)
J1-6	(J2-10)	CK	Digital (in)	Serial clock signal (+5V CMOS logic levels)
J1-7	(J2-9)	/BUSY	Digital (out)	Active low, Open-collector signal to indicate module is accessed
J1-8	(J2-8)	SRFB	Analog (out)	Sub-band S-meter and relative TX power for
J1-9	(J2-7)	SRFA	Analog (out)	Main-band S-meter and relative TX power for
J1-10	(J2-6)	DETB	Analog (out)	Sub-band RX audio (no de-emphasis), 90mVrms
J1-11	(J2-5)	DETA	Analog (out)	Main-band RX audio (no de-emphasis), 90mVrms
J1-12	(J2-4)	SQ2B	Analog (in)	Sub-band squelch adjust (DET signal via 5K pot to GND)
J1-13	(J2-3)	SQ2A	Analog (in)	Main-band squelch adjust (DET signal via 5K pot to GND)
J1-14	(J2-2)	SQSB	Digital (out)	Sub-band squelch gate (>3.5V = open, <1V = closed)
J1-15	(J2-1)	SQSA	Digital (out)	Main-band squelch gate (>3.5V = open, <1V = closed)

Note: all digital signals are +5V CMOS logic.

Table 1. Front Unit J1/2 daisy-chain connector pinouts. Signal directions (in or out) are from the standpoint of the UX module.

© Joseph M. Haas, 2025, all rights reserved

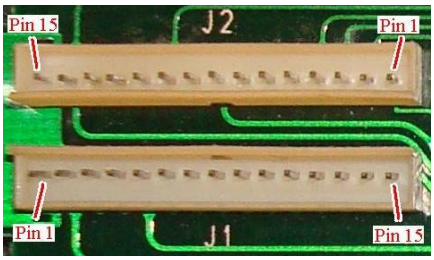
Expansion daisy-chain bus (J3 is lower, J4 is upper):

Pin#		Signal Name	Type	Description
J3-1	(J4-15)	PTT	Digital (out)	PTT engaged when <1V
J3-2	(J4-14)	TMUTE	Digital (out)	>2V when UX-S92 PLL unlocked
J3-3	(J4-13)	KEY	Digital (in)	Base controller TX key signal (+5V keys 2M SSB)
J3-4	(J4-12)	BAND6	Digital (in)	Band 6 select signal (UX-R91, WBFM Receiver)
J3-5	(J4-11)	BAND5	Digital (in)	Band 5 select signal (module association unknown)
J3-6	(J4-10)	BAND4	Digital (in)	Band 4 select signal (UX-S92, 2M SSB)
J3-7	(J4-9)	BAND3	Digital (in)	Band 3 select signal (module association unknown)
J3-8	(J4-8)	BAND1	Digital (in)	Band 1 select signal (module association unknown)
J3-9	(J4-7)	UNLKV	Digital (out)	UX-S92 PLL Unlock when >2V
J3-10	(J4-6)	UNLKU	Digital (out)	UHF PLL unlock signal (module unknown)
J3-11	(J4-5)	UNLK12	Digital (out)	1.2GHz PLL unlock signal (module unknown)
J3-12	(J4-4)	RITST	Digital (in)	RIT strobe (see text)
J3-13	(J4-3)	SSBST	Digital (in)	SSB strobe (see text)
J3-14	(J4-2)	PLST	Digital (in)	PLL strobe (see text)
J3-15	(J4-1)	CTRST	Digital (in)	Control strobe (see text)

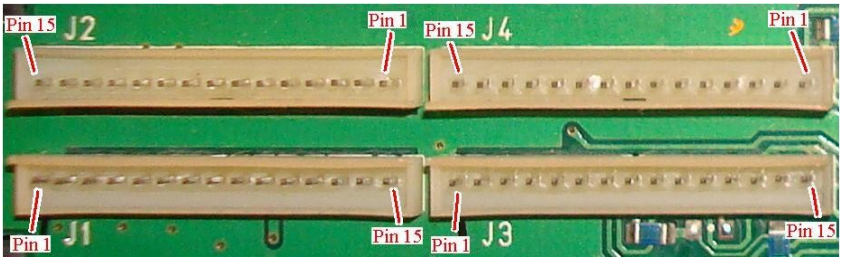
Note: all digital signals are +5V CMOS logic.

Table 2. Front Unit expansion connector pinouts. Signal directions (in or out) are from the standpoint of the UX module.

Tables 1 & 2 list the signals that are available on the Front Unit daisy-chain connectors. A close-up view of the connectors for the FM and expansion modules is shown in Figures 3a and 3b. The pin numbering reversal depicted in Table 1 occurs because ICOM rotated the top connector (J2 or J4) 180 degrees from the orientation of the bottom connector (J1 or J3) while connecting the pins directly between the pair of connectors on the bottom of the PWB. This rotation can cause no small degree of confusion if one is trying to design a custom Base Unit controller.



(a)



(b)

Figure 3. Close-up views of the daisy-chain connectors, (a) is the layout of an FM-only module, (b) is the layout of the UX-91/92. When stacked vertically, J1/2 of the FM-only modules will line up with J1/2 of the UX-91/92 modules

The answer to "Which pin-1 to use?" depends entirely on where the Base Unit is to be located. A base unit on the bottom of a UX module stack would orient pin 1 of the corresponding connectors to match that of the top connector of the Front Unit (J2 or J4). A top-mounted control unit with UX modules underneath would orient pin 1 to match that of the bottom connector of the Front Unit (J1 or J3). A controller in the middle (not recommended) would need two connectors oriented in the same fashion as the standard ICOM UX module Front Unit.

© Joseph M. Haas, 2025, all rights reserved

A common issue with the UX modules is that the OPC-179 cable is damaged or missing. A close mate to the module male housings is the Molex 87439-1500 female housing and the 87421-0000 crimp pins. Using the Molex parts and suitable wire (26 or 28 AWG stranded) replacement cables can be fabricated by cutting 15, 1.875” long wire pieces, strip at each end (0.06”), and apply female crimp pins to each wire end (sounds easy enough, except that the crimp pins are extremely small, and the approved crimp tool is very expensive). These 15 wires are then inserted into the two female housings to create the UX daisy-chain cable as shown in Figure 4. The trick is to get the housings oriented properly. I’ve always referred to these items to 1:1 cables, but the truth is that they are crossover cables because the wires do not connect the same pins on the two housings. The wire map is pin A1 (pin 1 of housing A) connects to pin B15 (housing B), pin A2 connects to B14, A3 to B13, etc…. When you lay the OPC-179 out flat, it appears to be wired “1:1” but you have to look closely to notice the crossover effect. Because the top and bottom connectors on the Front Unit are reversed from one-another, the cables end up connecting the signals from one module to the next in “straight-through” fashion.

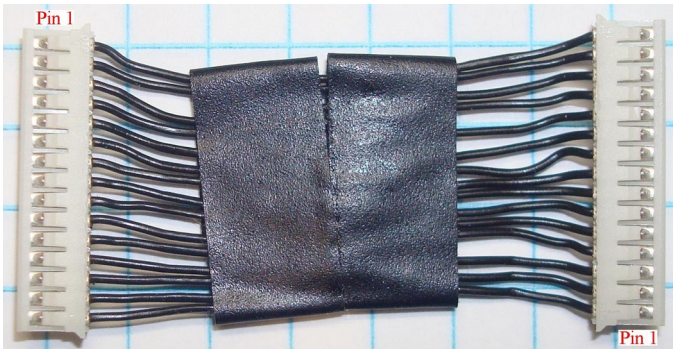


Figure 4. OPC-179 “joint” cable

The J1 through J4 digital signals require +5V CMOS logic. There are no bi-directional signals, so this simplifies the microcontroller (MCU) interface. A controller that is designed for FM only UX modules is much simpler since the additional control signals of the expansion bus are not required. Figures 5a and 5b illustrate simplified block diagrams for an FM-only, and a full-featured UX module controller. The MCU is typically a single-chip device which contains all of the I/O and internal peripherals needed to provide the signals and timing needed to access the UX modules. However, any microprocessor-based computer will work if it has sufficient I/O signals available and enough processing power to handle the serial data transfers.

The command port is depicted as a serial connection, but this could be a direct user interface if the MCU has enough additional I/O to drive a key-matrix and display. The UX module operates with 5V logic levels, so modern 3.3V (or lower) MCUs will likely require level shifting circuits (not shown). A simple level shifter for 3.3V to 5V logic is the 74HCTxx family with a 5V supply (the 74HCT1G08 is a good choice for a non-inverting level shifter). These devices operate on TTL voltage levels inputs, which are approximately equal to 3.3V CMOS levels, but output 5V CMOS levels (which are compatible with the UX module logic circuits) and therefore offer a simple level translation option for 3.3V processors.

© Joseph M. Haas, 2025, all rights reserved

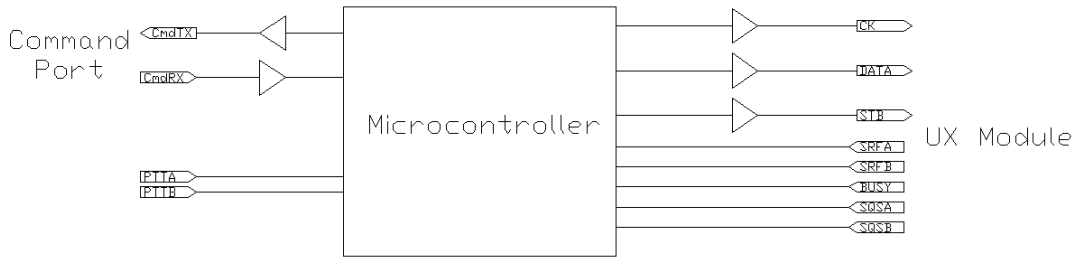


Figure 5a. Block diagram of an FM-only controller

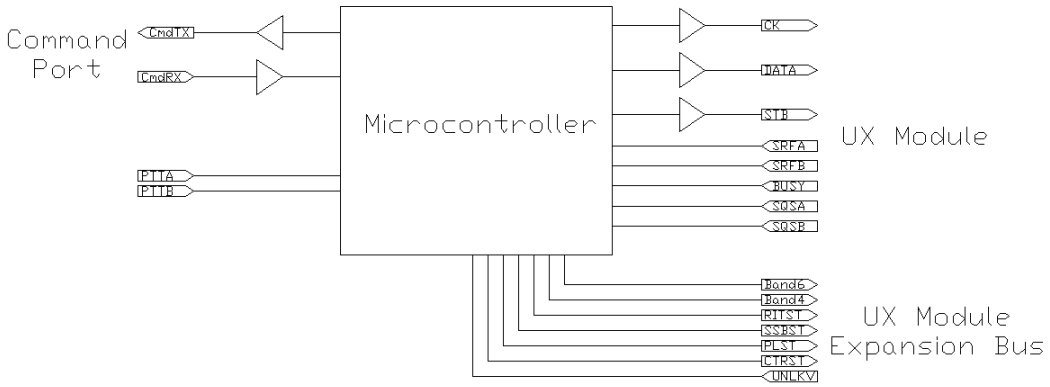


Figure 5b. Block diagram of a full-featured controller

Any remote-control interface for the UX units will be highly software dependent and this document does not directly address the software needed to control the UX modules. The programming language and processor type would be determined by the designer and a discussion of those choices is beyond the scope of this document. From a software requirements standpoint, the MCU should be able to transmit the serial data with timings that approximate the ICOM implementation, with enough I/O to drive all of the relevant control signals. The interface signals resemble standard SPI and UART interfaces, but the bit widths (20 bits or more) make it unlikely that a MCU will have a built-in peripheral that is ready-made to accommodate the signals in question. Higher-end MCUs will tend to have more advanced peripherals, and I have had luck configuring SPI peripherals on a TI ARM MCU to accomplish the long-bit transfers with minimal processor overhead.

In addition, the controller software must be capable of handling the transitions from receive to transmit and vis-à-vis, so it would need a PTT input signal (two if separate sub-band transmit control is desired). Also, if the controller is to access the SRF signals, an analog to digital converter is required (two if main/sub band architecture is implemented). The speed of the A/D conversion is not terribly critical for most applications, and suitable performance possible with as little as 8 bits of resolution (if the A/D span closely matches the desired input range of about 2.0 V).

The UX module analog signals require some attention to detail but need not be terribly complicated. At the bare minimum, the modulation audio provided to the modules on the MOD pin must be pre-emphasized (a response slope of +6 dB/octave across the voice band) and band limited to 3 KHz. Amplitude limiting (i.e., clipping) is recommended to prevent the module(s) from over-deviating on signal peaks. Limiting may be omitted if the source audio is already amplitude limited (such as a PC audio or modem source). High-pass filtering of the CTCSS band (with an F_c of around 300Hz) is also recommended for applications where CTCSS encode features are implemented.

© Joseph M. Haas, 2025, all rights reserved

A diode limiter for a 0Vdc offset pre-emph signal is shown in Figure 6. The diodes are offset, or biased, such that the limit levels are about 5.8 Vpp. The reason for this is that the diode limiter response is generally "mushy" in that there isn't a hard limit at 0.7V – this makes it difficult to get a good deviation limit since the deviation can continue to increase long after severe distortion is realized. By applying bias, the limit level is expanded, but the diode response remains the same. Following the modified limiter with a voltage divider reduces the degree of the "mushy" diode response by an amount that is proportional to the voltage division ratio. Another option for this circuit is to utilize an op-amp with a power supply voltage that is selected so that the maximum output swing of the op-amp defines the limiter range (generally, this op-amp is given a separate power supply that is designed to match up with a particular op-amp performance specification). The output of the op-amp is then fed to a resistive voltage divider to reduce the peak-to-peak amplitude of the signal.

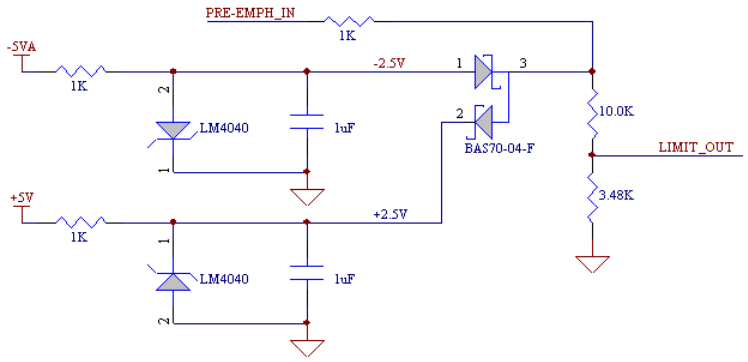


Figure 6. Modified diode limiter schematic

A properly aligned UX module will need 352 mVpp (125 mVrms) at the MOD pin for 5 KHz of deviation. Consequently, a CTCSS tone would need to be 50 mVpp (18 mVrms) to provide 0.75 KHz of deviation. The controller should provide a deviation control that can vary the modulation output from about 750 mVpp down to zero. This would put the nominal adjustment point at the mid-range of the pot swing. Figure 7 summarizes the necessary building blocks for a complete modulation shaping circuit for the FM modules.

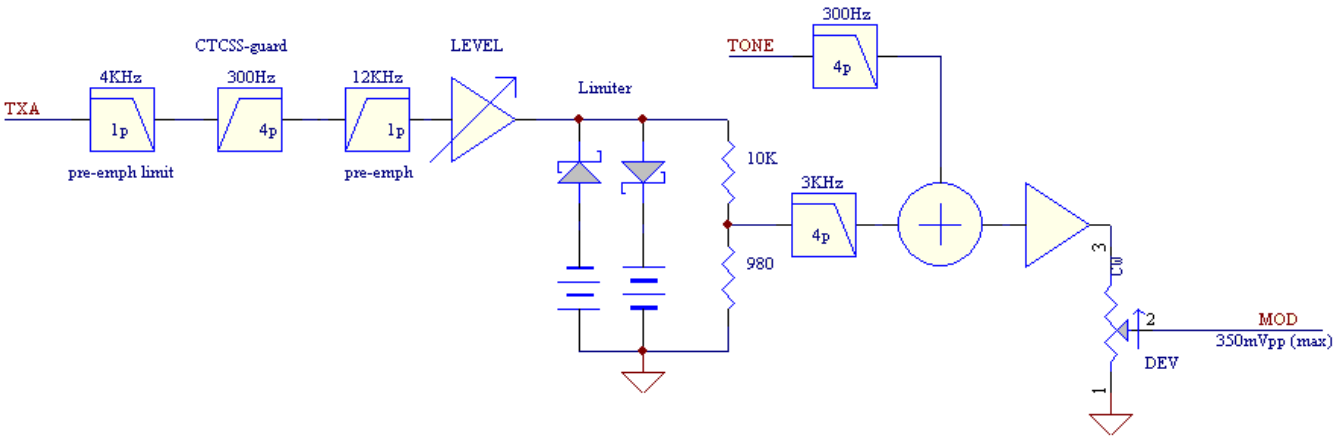


Figure 7. Block diagram for the FM MOD filter chain for an ICOM UX module

© Joseph M. Haas, 2025, all rights reserved

When including the UX-S92 module in the modulator design, one must take into account the fact that the SSB shaping requirements are completely different than those of the FM modules. The pre-emphasis and CTCSS guard blocks are not used, and the limiter is also not required (a compressor feature generally needs to be operator adjustable which is no the case for FM limiting). Generally, it is easier to produce two modulation shaping circuits, and switch between them at the MOD pin node. This makes it easier to address the widely different modulation requirements that exist between FM and SSB.

For reception, the DETA/B signals provide raw audio from the demodulator. De-emphasis is needed and can be provided by a simple RC low-pass filter with an Fc of 150Hz or less, as shown in Figure 8. A properly aligned UX module will produce 0.255 Vpp (90mVrms riding on a 3.5 Vdc offset) at the DET pin when receiving a 1 KHz tone with 3 KHz deviation (full quieting). In addition, the squelch threshold circuit must be satisfied with an adjustable resistance of 0 to 5 KΩ which sets the squelch level threshold. This resistance may be in the form of a traditional pot or may be implemented with a digitally adjustable pot with a maximum allowed voltage of at least 5V. The squelch output (SQ2) is a COS (carrier-operated-squelch) indicator only, the DET output is not muted when the squelch is "closed" which implies that a mute circuit for each DET signal is also needed for general purpose applications. Either the CD4066 or CD4053 analog switch ICs are two device choices that can be easily implemented as audio squelch gates.

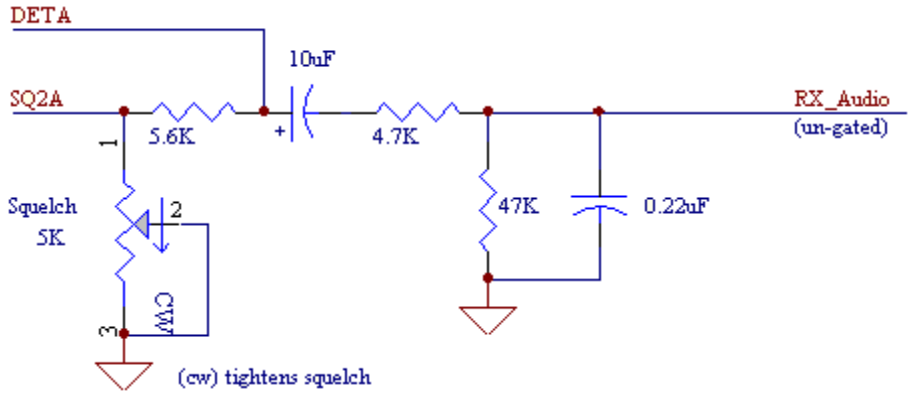


Figure 8. Simple RX Audio & squelch interface (From ACC's FC-900 design)

The last analog signal to be addressed is the SRFA/B signal. This signal is used to indicate S-meter reading when receiving, and RF power output when transmitting. If the SRF features are not desired, these pins may be left un-terminated. During receive, the voltage for SRF varies from approx. 0.02 V for no-signal, to approximately 1.5 V for a full-strength signal (however, the exact maximum value will vary somewhat between modules).

After observing the behavior of the SRF signal during transmit and carefully analyzing the circuit that generates this signal, I have concluded that this signal is simply an indicator that the module is producing measurable power. The op-amp stage that drives the SRF signal generally has better than 30 dB of excess gain which results in saturation of the op-amp, and no real indication of the relative power except that it is above some minimum level (my analysis indicates that this power threshold could be as low as 10mW). One might challenge this observation by noting that the ICOM control head indicates different SRF meter readings at low power vs. high power. However, this is simply a vestige of the controller software distinguishing between the high and low power setting and adjusting the display accordingly. I was able to prove this postulation by placing a variable resistor from SRFA to ground on a UX module. This resistance modified the SRFA voltage by reducing it as the shunt-resistance was

© Joseph M. Haas, 2025, all rights reserved

reduced. When the SRFA voltage reached about 0.32V, the SRF indication on the IC-901 control head would disappear entirely, without any gradual transition. The module was still producing rated power, but there was no indication on the display.

It is possible to modify each module by removing a resistor (the feedback resistor of the op-amp, e.g., R132 in the UX-19) and replacing it with a different value (or a pot) to reduce the gain of the SRF op-amp to the point that the SRF signal will provide a meaningful indication of the RF output power. This would require each module to be modified and would likely affect the operation of the transmit indicator if the module were to be used with the IC-900 or IC-901 radio systems. However, the change would be relatively easy to accomplish and would not affect any of the other module systems. The aim is to reduce the gain of the SRF op-amp to a value of about 4 (for most of the modules the gain of this stage is in the range of 67 or higher). The exact values might require some experimentation to get the gain to the point where it is usable, providing an SRF voltage of about 1.5V for full power. A pot adjustment would be advisable to allow multiple modules to be calibrated against each other. The reading would still need to be taken with a grain of salt since it will depend on the RF power produced by the module as well as the antenna match but might prove to be a simple and helpful tool for remote installations.

The SRF signal for the UX-S92 is different from that of the FM modules in that it produces an SRF signal that **IS** proportional to the instantaneous power and varies from 0V to 1.2V (matched load, high power – to be safe, one should assume that this voltage could vary to as much as 2.5V for full power into an infinite SWR). However, the proportionality constant is not linear, it is square-law. This is because of how the module produces the SRF voltage. RF is sampled from the final transmit matching chain and rectified, then it is filtered and buffered. By ohms law, $V^2/R = P$ and thus, $V(SRF) \propto \sqrt{P}$ and, assuming that the antenna match is near perfect, the proportionality constant can be calculated by measuring V(SRF) for several (at least two) power output levels. Using the UX-S92 module I had available, the following equation was determined by measuring V(SRF) at two different (and measured) RF power levels and averaging the result. Thus, $Pout = (V(SRF)/0.225)^2$. The proportionality constant is likely to vary somewhat from one module to the next and, in this case, is only valid for an RF $R_L \approx 50\ \Omega$.

The FM-Only Front Unit

The FM-only version of the ICOM Front Unit is shown in Figure 10. The expansion Front Units are discussed separately later in this document. The Front Unit serves two basic functions: First, it serves as an electronic, two position switch for the receiver signals to be selected as either the main or sub band connection to the base unit. Second, it holds address decode and steering logic for the PLL serial connections. The address decoding also captures several control signals to set the module's receiver switch (to select main or sub) and any logic signals that are particular to the module at hand.

Figure 10. ICOM UX module "Front Unit" (FM modules)

The receiver switching is handled by a series of CD4066 analog switches that are wired as four SPDT switches that connect the receive signals to either the "main" or "sub" buses that run via the 15 pin

© Joseph M. Haas, 2025, all rights reserved

OPC-179 cables. There are four pairs of signals that are switched: SQ2A/SQ2B, SQSA/SQSB, DETA/DETB, and SRFA/SRFB. Logic signals from the address decode logic select the A (main) or B (sub) connections to the receiver circuits. Because the UX stacking protocol only features a single MOD signal, the only switching involved is an on-off transistor. If one implements a system whereby the main and sub-units can both transmit, they must transmit the same audio unless modifications are made to the Front Unit daisy-chain scheme to allow for separate modulation audio paths.

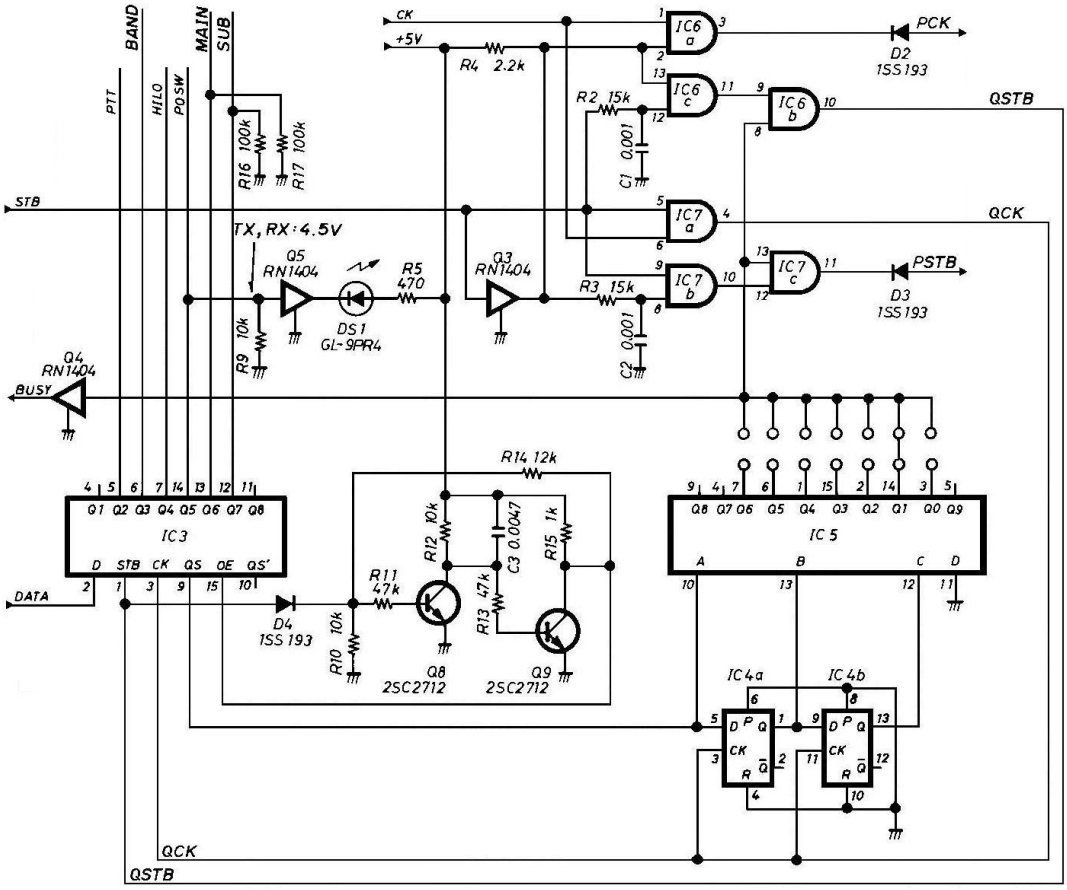


Figure 11. ICOM UX address decoding and steering logic circuit.

Figure 11 depicts the decoding and clock steering logic of the FM-only Front Unit. The steering logic is accomplished by a combination of a 10-bit shift register (comprised of a CD4094 8-bit SR, IC3, a CD4013 dual D flip-flop, IC4, and a CD4028 one-of-8 decoder, IC5) and a series of logic gates.

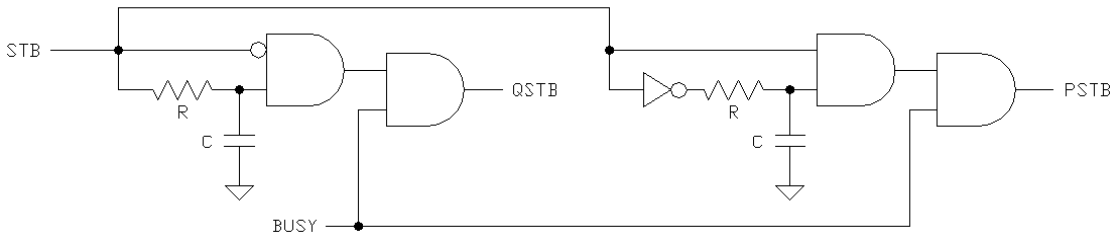


Figure 12. Simplified QSTB & PSTB edge detector circuits

© Joseph M. Haas, 2025, all rights reserved

A simple edge detector constructed from an RC network and an AND gate is used to produce pulses at the rising or falling edge (depending on the configuration of the gate and RC combination) of the strobe (STB) pulse. In this manner, the PSTB and QSTB signals are produced. PSTB drives the PLL strobe, while QSTB drives the IC3 strobe to latch the module control signals.

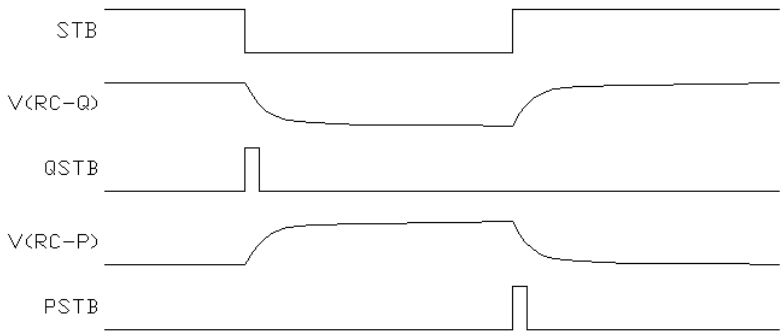


Figure 13. QSTB & PSTB circuit waveforms.

Figures 12 and 13 depict the operation of the QSTB and PSTB edge detector circuits. The QSTB pulse is produced by delaying an inverted version of the input signal (using an RC time constant) and feeding both signals through an AND gate. The width of the pulse is proportional to the value of RC (15 KΩ and 0.001 µF) and the switching threshold of the AND gate. Using the equation for RC decay (where V(RC) is normalized), $V(RC) = 1 - e^{-t/RC}$, and solving for t gives $t = -RC\ln(V(RC))$. If one assumes a 50% gate switching threshold for V(RC), the values of R and C used in the UX modules result in a pulse width of about 10µs.

The switching threshold of the logic gate is the most nebulous of values in the above equation, varying with the IC processes and temperature. At the low extreme of 80% for the switching threshold (a typical datasheet minimum), t = 3.3µs which is actually a bit low for some of the PLL devices. Room temperature measurements of PSTB give a value of 10.3 µs (down to 9.8 µs when the AND gate was exposed to a blast of freeze spray). A better choice of gate for this scheme would be a Schmitt-trigger device (such as the SN74HCS08DYYR) which provides more consistent voltage levels for high and low levels, and reduces the change of noise in the transitions. The following table summarizes some of the specifications of the PLL devices used in the UX modules (the IC-901 base unit devices are listed for reference):

Part#	**UX Module**	**T_{low} (°C)**	**T_{high} (°C)**	**t_{PSTB} (min)**	**Bit Order**
PLL2001	UX-19, UX-59, VHF†	-30	+80	2.7µs	msb first
MB87001A	UX-29, UX-39	-40	+85	1µs	msb first
µPD2834C	UX-49	-40	+85	5µs	msb first
TC9181P	UX-129, UHF†	-30	+85	2.7µs	lsb first

† IC-901 Base-Unit "modules"

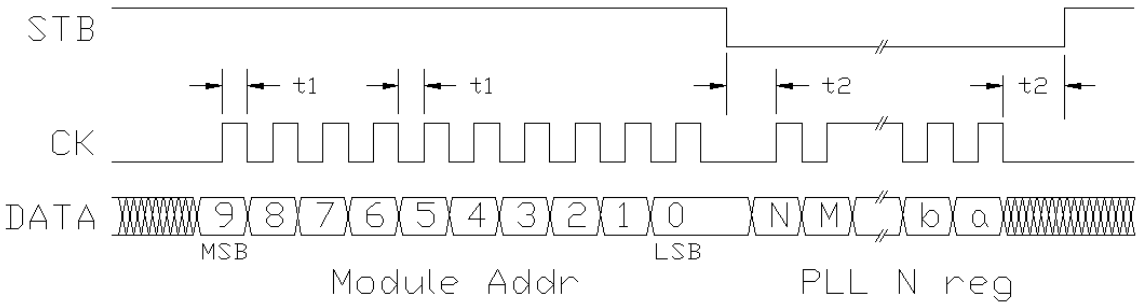
Figure 14 illustrates the bit-fields and critical timing to send data to a UX module. IC3 is a buffered shift register – the shift register section is always active, and data clocked through the register is shifted back out on pin 9. This output of IC3 and the two flip-flops of IC4 form a 3-bit shift register (un-buffered) that captures the module address bits, while the remaining 7 bits of IC3 capture control data. When the address bits (IC4 pins 5, 1, and 13) match the device address that is selected by the solder jumper adjacent to IC5 (only one of the 7 jumpers may be soldered at a time), the BUSY output is

© Joseph M. Haas, 2025, all rights reserved

activated which enables the PSTB and QSTB signals. The BUSY signal is also connected to the base unit and allows the controller to determine if an addressed module is installed at power-on.

STB is normally high and is brought low at the end of the 10-bit module control/address frame. This readies the steering logic for PLL data and clock and pulses the QSTB line to latch the module control signals. When the QSTB goes high, the IC3 shift register bits are transferred to the IC3 output pins and these pins retain their value when QSTB goes low again which latches the module control signals. I apply an "edge delay" of at least 150µs after the STB edge to make sure that the steering logic has been properly engaged, but any delay longer than 20 or 30 µs should be sufficient.

Next, PLL data can be sent. The ICOM IC-900/901 radios send PLL frames as fixed 20-bit wide values, but each PLL type has different register widths and so the fixed 20-bit width is not necessary (ICOM likely sends a fixed 20-bit frame to simplify the control hardware & software). In addition, the shift order of the PLL data is not the same for all modules. Most of the modules shift msb first, but the UX-129 & IC-901 UHF PLL data shifts lsb first. The module control/address data is always shifted msb first. Whatever the shift direction, the key is to align the bits so that any "extra" bits get shifted first. Once the PLL clocking is finished, the controller raises STB high and the next transaction can occur (after waiting the PSTB edge delay).



t1	clock half-period duration, 104 us (min)	t2	strobe settling time, 150 us (min)
D[9:7]	module address (bit 9 = first bit to shift, msb)		
Module Control Signals, D[6:0]:			
D6	main bus select	D3	RF power ("1" = high, "0" = low)
D5	sub bus select (D[6:5] = "11" is not allowed)	D2	BAND signal
D4	module power on	D1	TX enable (PTT3)
		D0	not used

PLL Data:
D[N:a] PLL data (20 bits, typ. "N" = first bit to shift, msb for all except UX-129)

Figure 14. UX Module serial data diagram.

Serial data to be sent to the module is clocked at the rising edge of CK which should have an edge-to-edge half-cycle period of 104µs to match the ICOM timing (this is very conservative compared with the device specifications, but is necessary due to the ICOM circuitry and physical layout of the module stack). To gain the maximum noise margin, data should transition at or near the falling edge of the clock to allow the signal to settle at the far-end of the signal path before asserting the rising edge of the clock. The ACC controller used open drain/collector style drivers to clock data into the band modules, but push-pull drivers are a better choice, especially if they are bus-driver devices (like the 'HC244). Push-pull devices actively assert the signals in both the high and low states giving rise and fall times that are more consistent and faster than open drain/collector devices. Open drain/collector drivers are actively asserted only in the "0" state. In the "1" state, the signal level is established with a pull-up

© Joseph M. Haas, 2025, all rights reserved

resistor. The rise time of the signal is then determined by the value of the pull-up resistor, and the stray capacitance from the signal to ground, while the fall time is determined by the R-on of the driver and that same line capacitance (R-on is generally many orders of magnitude smaller than R pull-up).

The PLL data varies depending on the PLL chip and module architecture. These differences are covered in the Main Unit section below. In addition, the BAND signal is not used except on the 2M and 6M FM modules, and it is also covered in the Main Unit section.

There is an aspect of the PLL data transfer that I have not explored as yet. It would greatly reduce the amount of time required to configure some of the PLL devices. For those PLL devices that require initialization frames, STB can be pulsed high for, say, 50 µs in between each PLL register. This would reduce the number of bits (by a factor of 10 bits per register, since the control/address bit frame would not need to be re-sent) that were required under the ICOM method. This is especially attractive for the UX-129 since there are 4 different registers that must be configured for the TC9181P PLL. If CK is low, STB can be pulsed to trigger PSTB. The STB pulse should be no less than 10 µs, and there should be at least 50 µs between STB pulses.

The Main Unit Module Interface

The Main Unit contains all of the receiver, transmitter, and power supply circuits. The discussion here focuses on the behavior of the interface signals. These are divided between the serial data signals, and the analog TX/RX signals. All of the modules use a common interface for these signals with the primary difference between modules being the format of the PLL serial data. The analog signals have been covered in the previous sections.

Three signals are provided to the Main Unit for the serial data interface to the PLL: DATA, CKD, and PSTB. In addition, the POSW, PTT3, and BAND signals are provided to control the Main Unit circuits. POSW is set to 1 to enable the Main Unit power supply and thus turns on the UX module. When PTT3 is set to logic 1, the TX circuits are enabled which puts the module in transmit mode. *Note: The PLL must first be configured for the desired TX frequency* ***before*** *PTT3 is activated or the module will produce RF output on an out-of-band frequency.*

BAND is used by the UX-59 and UX-29 modules to bypass the receiver front-end bandpass filter. This increases the receiver sensitivity outside the specified (Ham band) frequency range to allow reception of out-of-band signals. Of course, this increase in sensitivity comes at the cost of selectivity, so the user must be aware that they may have to tolerate an increased level of co-channel or IMD interference when receiving signals in this configuration. BAND is normally set to 0 for ham-band coverage and is set to 1 to enable out-of-band coverage. Even with the filter bypass, the out-of-band sensitivity generally leaves something to be desired (the 12dB SINAD sensitivity can be as much as 5 or 10 µV or higher) but it is much better than with the in-band filtering enabled and can still be useful for receiving fixed service stations such as NOAA weather broadcasts.

It should be noted that the ICOM UX modules all feature a single PLL that serves double duty as transmit and receive LO (most modules operate their VCO at the TX operating frequency). The receiver IF sets the VCO frequency difference between RX/TX and this is generally different for each module type. This means that the control system must send PLL data at least twice for every transmit/receive cycle (once to set the TX frequency on initial transmit, and again to set the receive frequency when the PTT is released).

© Joseph M. Haas, 2025, all rights reserved

PLL Basics

There are several factors that go into the design of a PLL system and a full discussion of these factors is beyond the scope of this document. Because of this, only the factors that relate to operating frequency are discussed here (and even at that, the discussion is very light). Even though some of the UX modules share the same PLL device, the architecture of the receivers is such that none of the modules share the same exact PLL-N and ref-divider code conversion algorithms. Knowing something about how the PLL works and how the module architecture is arranged can greatly simplify the understanding of the PLL control algorithms. However, this understanding is not necessary to make use of the information. Some of these architecture and circuit specifics are discussed below, but in the end, one only needs to know a base PLL pattern, base operating frequency for RX and TX, and bit shift order in order to configure the module's PLL (these items are all tabulated below).

The simplest PLL is nothing more than a programmable digital divider/counter chain that accepts the output of a VCO (voltage-controlled oscillator). The final carry output of the PLL divider/counter is fed to a phase comparator (a basic phase comparator is nothing more than a two-input exclusive-OR gate). The other input to the phase comparator comes from a reference oscillator. The output of the phase comparator is filtered and fed back to the VCO input. If the divided VCO signal and reference oscillator are exactly in-phase and on-frequency, the output of the phase comparator will be zero (or some base-bias value, depending on the design of the phase comparator). Any phase difference will result in a non-zero phase comparator output which, once filtered, becomes an error voltage that drives the VCO to return the system to a zero phase-error.

Some PLL devices have built-in reference oscillators, while others require an external reference oscillator. Additionally, some PLL devices have a separate reference divider that is programmable. Most PLL devices also have provisions for dual-modulus prescaler devices that divide-down the VCO output so that the PLL logic can process the signal (most PLL devices can only accept VCO inputs up to a couple of hundred MHz). The dual-modulus architecture basically allows fractional division of the VCO to allow small frequency steps (e.g., 5 KHz) to be achieved. These prescaler/PLL combinations can cause the PLL divider to be split into two dividers, which must sometimes be dealt with in the PLL register conversion logic when calculating a PLL register setting from a desired operating frequency value (e.g., bit stuffing the divider value).

Finally, some PLLs feature still other configuration registers which can affect the configuration of the phase comparator or allow control of general purpose I/O signals (as is the case with the PLL used on the UX-129 & IC-901 UHF).

The reference oscillator is one of the controlling factors in the determination of minimum frequency step in the VCO output. Typically, the ICOM PLLs operate off of a 5 KHz base reference which is derived from a 5.12 to 12.8 MHz oscillator (depending on the particular module) and a reference divider that is configured to produce a 5 KHz reference. If the PLL has a reference divider, this must be configured before PLL data can be sent.

All of the PLL devices used in the UX modules can be configured once at power-up and those configurations will be retained until later modified, or power is removed. However, the ICOM protocol is to set all of the configurations every time the PLL "N" divider is modified. This is a fail-safe method which ensures that the PLL will at least recover to normal operation on the next frequency change if its configuration is corrupted. The overhead for configuration is not excessive if the ICOM clock timings are used so I haven't tried to investigate how the modules might perform otherwise.

© Joseph M. Haas, 2025, all rights reserved

PLL Calculations

From my earliest investigations into the IC-901 behavior, I developed a means of calculating PLL values based on the empirical bit patterns that I had observed. Now that I've studied the actual PLL device datasheets and have a greater understanding of the PLL bit definitions, little has changed regarding how I manipulate the bits for frequency control, but now I can see why they work and other options for how the same result can be achieved.

UX Module	Base Freq.	Base PLL (hex)	TX offs (hex)	Mod bit	PLL Init	Bit Order
UX-19	28.000 MHz	0x01e3b	(-)0x085b	none *	0x1325	msb first
UX-59	40.000 MHz	0x02a2e	(-)0x0aee	none *	0x1325	msb first
UX-29	136.000 MHz	0x05cd0	(+)0x0d70	6	none	msb first
UX-39	220.000 MHz	0x11e70	(+)0x0d70	6	none	msb first
UX-49	400.000 MHz	0x1266a	(+)0x1216	none	none	msb first
UX-129	1200.000 MHz	0x19f64	(+)0x355c	6	† see notes	lsb first
901 UHF	400.000 MHz	0x12061	(+)0x181f	6	†† see notes	lsb first
901 VHF	136.000 MHz	0x05cd0	(+)0x0d70	none *	0x01401	msb first

† UX-129 has several initialization frames:

Ref osc init:	0x0500	(20 bits)
"HL" init:	0x0	(only 4 bits needed)
"GPIO" init1:	0xc	(only 4 bits needed)
"GPIO" init2:	0xf	(only 4 bits needed)

UX-129 data is shown in normal, msb-left format,
it must be sent to the PLL in bit-reversed, lsb first format.
Register order = REF OSC init; PLL N reg; HL Init; GPIO-2 init; GPIO-1 init
1296 bit pattern calculated for Fop/2 (w/5 KHz step) and is doubled to get Fop with 10 KHz step.

** These PLL registers must be left shifted one bit to align the divider bits after all PLL calculations have been performed*

PLL_N (RX) = $((F_{op} - F_{base})/Fstep) + Base\_PLL$
PLL_N (TX) = $((F_{op} - F_{base})/Fstep) + Base\_PLL + Txoffs$
Fstep = 5 KHz for all but UX-129 which is 10 KHz

†† 901 UHF has several initialization frames (uses same PLL as the UX-129):

Ref osc init:	"10b" + "ab" + 0x0A00	L2/L1 ("ab") = "00b"
PLL:	"01b" + <pll calc>	"0" inserted at bit 6
"HL" init:	"00b" + "00b"	
"GPIO" init1:	"11b" + "ab"	b = RX filter control 0 = normal, 1 = bypass). a = not used

901 UHF data is shown in normal, msb-left format, it must be sent to the PLL in bit-reversed, lsb first format.
Register order = REF OSC init; PLL N reg; HL Init; GPIO

Table 3. ICOM UX and IC-901 PLL Register details

The basic idea is this: one starts with a base PLL value (a 20-bit HEX value) that corresponds to a specific RX frequency (the base frequency) which is used to calculate the frequency difference between the desired frequency and the base RX frequency. Since the PLLs all operate with Fstep = 5 KHz channel steps (the UX-129 PLL operates in 5 KHz steps, but the VCO is doubled for transmit and receive so the module uses an Fstep of 10 KHz for its calculations), the frequency difference is then divided by Fstep and the result is added to the base PLL value. If the desired frequency is for transmit, then a TX offset is added/subtracted, which is a fixed value for each band module that relates to the receiver IF frequency (TX offset = F_{IF} / Fstep). Some devices require bit stuffing to account for a prescaler, or bit reversal of the PLL value. The resulting value is then sent to the PLL "N" divider register to set the PLL operating point. Table 3 lists the various values needed for each of the UX modules.

The "Mod bit" column indicates if (and where) a '0' bit must be inserted into the PLL-N code to adjust the PLL divide code to account for a dual-modulus divider. For example, if bit 6 is the modulus bit (the

© Joseph M. Haas, 2025, all rights reserved

lsb is bit 0), a zero must be inserted into the bit pattern by shifting the PLL-N bits [20:6] one bit toward the msb, leaving bits [5:0] unchanged and bit 6 = 0. The PLL bits are arranged as 20-bit fields, but the various PLL devices actually use different register widths. The extra bits are set to "0" and simply shifted out of the PLL register. Note that the UX-129 requires the lsb be shifted first. This means that the application software must either bit reverse the value at run time (as would be needed if a MCU SPI interface were used), or a separate set of clocking functions would be needed for the UX-129 to clock the PLL data lsb first. I have implemented both versions for the UX-129 interface.

As an example, the PLL code for 52.525 MHz (UX-59) TX would be calculated as:

$PLL\_N_{reg} = (F_{op} - F_{base})/5000 + BasePLL + Txoffs$

`((52,525,000 - 40,000,000)/5000) + 0x02a2e – 0x0aee = 0x2909`

As indicated, there is no mod bit for the UX-59, but the entire PLL must be left shifted 1 bit to align with the PLL register (the right-most bit selects N-DIV or REF-DIV), which gives the final PLL-N code of `0x05212` to transmit on 52.525 MHz.

The Mechanics of PLL Updates

Just sending a single set of PLL updates isn't enough to properly operate the UX module unless it is only required to receive on a single frequency. Since the PLL must be adjusted for every TX/RX cycle, the system controller must be able to send the relevant PLL data whenever a transition is needed between RX and TX. Further, the operation of the PLL and VCO must be considered during these transitions. The main issue regards the step performance of the VCO/PLL combination. Since the VCO/PLL form a feedback control loop, step changes require some minimum settling time. During this time, the VCO frequency is rapidly changing and may experience over/under-shoot as the loop settles. This can be extremely undesirable in transmit mode, so steps need to be taken to prevent transmitting at an unknown frequency.

The best approach would be to monitor a lock signal from the PLL and engage the TX amplifiers (via the PTT3 bit in the module control field) after the PLL is in-lock. While the expansion modules (the 2M SSB module in particular) offer an un-lock signal that can be monitored by the control processor, the IC-900 series UX modules do not have the lock-status available. The only other alternative is to wait-in-the-blind for some period of time for the PLL to settle before activating the module PTT3 signal. Thus the protocol is to send the PLL data for TX with the PTT3 = 0, wait a certain amount of time for the VCO to settle, then send the PLL data again, with PTT3 = 1. The trick is in determining the settling time for the VCO. From empirical measurements of the IC-901, 10ms seems to be what ICOM uses for their PLL settling time.

One must be aware that there is a latency involved in the process of shifting from receive to transmit and vice-versa. The system requires a minimum of 60 bit-clocks plus 10ms to go from RX to TX (about 22ms), or about 12ms to go from TX to RX (this doesn't include any delay introduced by the processor in recognizing the transition between TX/RX). These latencies would be difficult to notice in an analog transceiver application but might require some attention for digital modes.

UX-S92 Module Overview

© Joseph M. Haas, 2025, all rights reserved

The UX-S92 “expansion” module is fundamentally different from the FM-only modules. Aside from the differences in the modulation, there are many different configuration registers for the UX-S92 vs. the two basic register types of the FM-only modules. The Front Unit for this module (shown in Figure 15) has a total of four OPC-179 cable connections to accommodate the additional interface signals required to facilitate the features offered by the UX-S92. While the basic serial transfers are similar to those of the FM-only modules, there are some important differences, as discussed below.

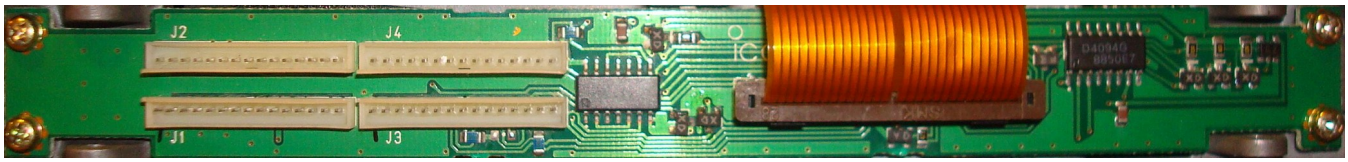
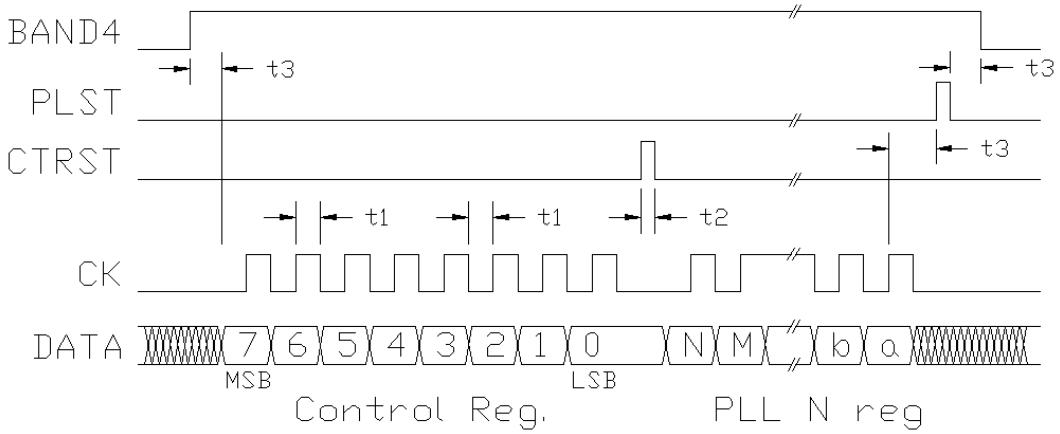


Figure 15. ICOM UX-S92 module “Front Unit”

The ICOM UX-S92 is a 2M SSB transceiver module and as such, there are several aspects of the module controls that are non-existent for an FM-only module. In addition, the modulation input to the UX-S92 is different from the FM-only modules (e.g., no pre-emphasis for SSB modulation) as was mentioned previously in this document.

For the UX-S92, the CK and DATA signals operate in a similar fashion as with the FM-only modules, however STB is not used. Instead, various strobe signals (CTRST, SSBST, PLST, and RITST) found on the expansion bus are used to strobe the various serial data formats. Figure 16 illustrates the serial transfer model for the expansion modules.



t1	clock half-period duration, 104µs (min)	t2	strobe width, 10µs (min)
D[7:0]	Control data (see text)	t3	clock setup/hold time, 104µs (min)
PLL Data:			
D[N:a]	PLL data (20 bits, typ. “N” = msb (first bit to shift)		

Figure 16. Expansion Module serial data diagram.

These transfers require the MCU to activate the appropriate band signal (BAND4 = 1 for the UX-S92) during data transfers and pulse the various strobe signals depending on the Main Unit device that is to be accessed (the pulses follow the last data bit transferred). The following table lists all of the controllable circuits that make up the UX-S92:

© Joseph M. Haas, 2025, all rights reserved

Control Description	Data Format (see text)	Strobe
Control data (CDAT)	8 bits, MSB first	CTRST
SSB control data (SDAT)	8 bits, MSB first	SSBST
PLL (coarse frequency control)	15 bits, MSB first	PLST (CDAT[1] = 0)
FDA (fine frequency control)	8 bits, MSB first	PLST (CDAT[1] = 1)
SDA (squelch)	8 bits	RITST (SDAT[7:6] = 01)
RDA (RIT/XIT)	8 bits	RITST (SDAT[7:6] = 10)

As shown above, PLST and RITST perform double duty acting as the strobe signal for one of two different registers based on the status of control data bits. This indicates that the control data transfers must be completed prior to loading these other resources. The control data registers produce logic signals that are used to control the SSB features and strobe steering.

Because this unit is AM based, the tuning is much more critical than for FM. This necessitates a much finer tuning step to than is available on the FM-only modules. Because of the difficulties in producing PLL circuit with increasingly smaller minimum steps, the UX-S92 adopted a multifaceted approach to managing the tuning step size. First, the PLL was designed to produce 5 KHz steps, and is controlled in much the same fashion as the PLLs used in the FM-only modules.

The architecture of the UX-S92 PLL is quite different, however, and features a reference oscillator and a down-conversion oscillator. The Reference oscillator establishes the 5 KHz frequency step, while the down-conversion oscillator is applied to the VCO output via a mixer and filter to produce the PLL-N divider input. The 5 KHz steps are then subdivided by tuning the down-conversion oscillator to produce an approximate minimum step of about 15 Hz using two separate analog tuning voltages produced by the FDA and RDA D/A converters.

The FDA converter is configured to produce 100, 50 Hz tuning steps to form the base minimum tuning step. The FDA is an 8-bit shift register that drives a simple R-2R ladder D/A that feeds the reference oscillator tuning circuit. Further refinement of the operating frequency comes from the RDA register. RDA is another simple 8-bit D/A circuit that also drives the PLL reference oscillator over a range of ±63 steps. However, the RDA adjustments result in RIT/XIT tuning steps of approximately 15 Hz increments due to the different scaling applied to the RDA signal applied to the down-conversion oscillator. Of course, the RIT tuning range overlaps the FDA range by a considerable degree, spanning almost 2 KHz.

The SDA is yet another simple D/A converter that sets the squelch threshold in 32 steps (as implemented in the IC-901). Even though this register has 8 bits, only the most significant 5 bits are used (except for step setting #32, which uses all 8 bits).

UX-S92 Control Protocols

There are several registers that require attention on the UX-S92 making it much more complicated to control than any of the FM-only UX modules. Some of these differences are obvious, while others can be more difficult to discern. However, the controls can be broken down into three distinct categories to simplify the control process: First, the module control registers must be asserted. Next, the operating frequency is adjusted, with the squelch control being last on the list.

UX-S92 Control Registers

© Joseph M. Haas, 2025, all rights reserved

The UX-S92 features two 8-bit control registers:

Bit#	CTRST (CDAT)	SSBST (SDAT)
7	mode1 (mode[1:0] = 00 is n/a, 11 is CW)	not used
6	mode0 (01 is LSB, and 10 is USB)	RIT select _____ Steers RTIST to RDA or SDA
5	If 1, main, else sub	SQU select __/ (these bits can not both = 1)
4	If 1, module power on, else off	If 1, slow AGC, else fast
3	If 1, RF power = high, else low	RFGa: if 1, RF atten = med. RFG[b:a] = 11 prohibited
2	not used	RFGb: if 1, RF atten = high. RFG[b:a] = 00, atten off
1	If 1, TX mode, else RX	If 1, Noise blanker on
0	If 1, steers PLST to FDA, else to PLL	If 1, RIT else RIT/XIT

Aside from the direct controls present in these registers, they each feature steering bits to control the routing of some of the strobe lines. In particular, the RITST strobe accesses the RIT D/A register (RDA) or the squelch D/A register (SDA) depending on the status of bits 5 and 6 of the SDAT register. Also, the PLST strobe accesses either the PLL (if CDAT[0] = 0) or the fine tuning D/A register (FDA) if CDAT[0] = 1.

The bit functions are relatively straightforward for most of the control bits. With the exception of the steering bits, the remaining controls are direct-action which is to say that the action of the bits is immediate and no other serial data transfers are required.

UX-S92 Operating Frequency

The operating frequency of the UX-S92 is more difficult to calculate because of the increased tuning resolution and the complicated nature of the tuning circuits. There are three separate register systems that each influence the operating frequency as previously discussed. Of course, the main influence is from the PLL registers, with the FDA and RDA registers offering decreasing levels of influence. The separated control of the operating frequency requires the controller software to break out the frequency calculations into 3 distinct results. It must calculate the 5 KHz PLL register values, followed by the 50 Hz step FDA register value, and finally, the 15 Hz step RDA register values. Other complications to these calculations derive from how to implement the RIT feature, and the fact that the modulation mode influences the frequency calculations.

The basic algorithm I have constructed starts with the operating frequency and adds an offset based on the modulation mode. Next, the 5 KHz PLL values are calculated. Finally, the remainder of the 5 KHz calculation is divided into 50 Hz and 15 Hz steps to arrive at the FDA and RDA register values. Once all of the calculations are made, these values are all sent to the module.

The UX-S92 PLL, when considered alone, operates much the same as the FM-only modules. It consists of a PLL chip and a VCO which establish the module operating frequency to a resolution of 5 KHz. The PLL chip, MC14515, features reference divider and N divider registers that are each 14 bits wide plus an additional bit (at the lsb position) to select the reference register, lsb = 1, or the PLL N register, lsb = 0. Adding the register select bit brings the PLL register transfers to 15 bits total and it uses an msb-first shift protocol.

The easiest way to handle the PLL is to use a 16-bit, unsigned integer to hold the binary value of the PLL register. When the PLL calculations are completed, shift the 16-bit register left one bit leaving the lsb = “0” for PLL N data, or set the lsb to “1” for reference (PLL init) data. Then shift the entire 16 bits,

© Joseph M. Haas, 2025, all rights reserved

msb first, to the PLL device. The following table illustrates the values used to control the PLL (Base PLL and PLL init are not right shifted 1 bit):

					MOD_offset (LSB = 0)	
UX Module	**Base Freq.**	**Base PLL (hex)**	**PLL Init**	**Bit Order**	**USB**	**CW**
UX-S92	139.9965 MHz	0x00d5	0x0400	msb first	+3 KHz	+2.2 KHz

The PLL-N register formula is: PLL_N = INT((F_{op} + MOD_offset – F_{base})/5 KHz) + Base_PLL. To calculate The FDA value, use this formula: FDA = ((F_{op} + MOD_offset – F_{base})/5 KHz) - INT((F_{op} + MOD_offset – F_{base})/5 KHz)) * 100. This is simply the fractional part of the 5 KHz remainder times 100. Finally, if one desires to calculate the nearest 15 Hz step, take the fractional part of the FDA result, and multiply by 50/15, the integer portion of which which becomes the RDA value (RIT or XIT must be enabled for RDA to be active).

My interface applications tend to utilize a VFO register that holds the operating frequency as packed BCD values or a long (32 bit) integer value. This drives my algorithms to work against the minimum frequency step resolution of the system with the RIT/XIT feature handled as a modifier to the VFO setting (i.e., the RIT/XIT is a separate register that is added or subtracted with the VFO when calculating the module register values). The IC-901, on the other hand, operates on a 50 Hz step, and the RIT/XIT feature is handled separately. In my opinion, which implementation to use is purely an esoteric issue.

UX-S92 Squelch

The squelch register shares the RITST with the RDA register. It is an 8-bit register that encodes the squelch value as a number between 0 and 32. The squelch value is encoded as 5 bits (0-31) and it is left shifted 3 bits so that the msb of the squelch value aligns with the msb of the register – the value for squelch position “32” is encoded as “1”s in all 8 bits of the register (0xff). Squelch values 0 – 31 have “0”s in bit positions 2:0 of the squelch register. The progression of squelch settings is thus 0x00, 0x08, 0x10, 0x18, 0x20, … , 0xf0, 0xf8, 0xff. A squelch value of 0x00 corresponds to fully open, while 0xff is fully closed.

UX-S92 Summary

The ICOM protocol generally sends PLL data first, followed by other control register data fields. Of course, the CTRST field is needed to send PLL data. The easiest way to approach controlling the UX-S92 is to group the control fields by function. The basic functions are Control Bits, Frequency, RIT/XIT value, and Squelch. Control bits are those bits contained in the CDAT and SDAT registers and include AGC, RF gain, noise blanker, RIT/XIT mode, modulation mode, module power, RF power, TX enable, and strobe steering.

The initialization frames for initial power would be as follows:

- Send CDAT with all zero bits.
- Send PLL N register data of all zero bits.
- Send CDAT with PLL steering selected and all other bits set as desired (including module power on).
- Send PLL init frame.
- Repeat CDAT and PLL init frames.
- Send CDAT with PLL steering selected and all other bits set as desired.
- Send PLL N-register frame for operating frequency.

© Joseph M. Haas, 2025, all rights reserved

- Send CDAT with FDA steering selected and all other bits set as desired.
- Send FDA register frame.
- Enable RDA steering in SDAT frame.
- Send RDA value frame.
- Enable SDA steering in SDAT frame.
- Send SDA value frame.

To change frequency:

- Send CDAT with PLL steering selected and all other bits set as desired (including module power on).
- Send PLL init frame.
- Send CDAT with PLL steering selected and all other bits set as desired.
- Send PLL N-register frame for operating frequency.
- Send CDAT with FDA steering selected and all other bits set as desired.
- Send FDA register frame.
- Enable RDA steering in SDAT frame.
- Send RDA value frame.

To change from RX to TX or TX to RX:

- Send CDAT with PLL steering selected and all other bits set as desired (including CDAT[1] set to the new TX status).
- Send PLL init frame.
- Send CDAT with PLL steering selected and all other bits set as in previous frame.
- Send PLL N-register frame for operating frequency.
- Send CDAT with FDA steering selected and all other bits set as in previous frame.
- Send FDA register frame.
- Enable RDA steering in SDAT frame.
- Send RDA value frame.

To change one of the radio controls (RF gain, AGC, etc…) simply send the SDAT or CDAT register as appropriate.

© Joseph M. Haas, 2025, all rights reserved

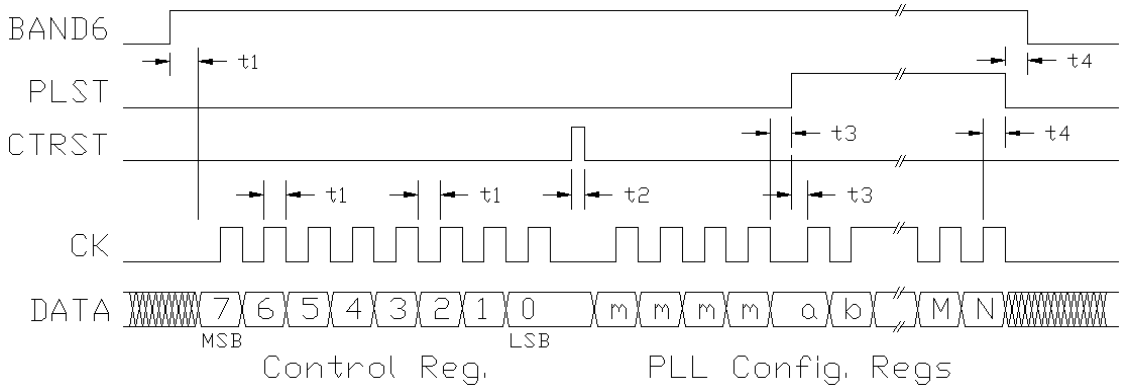
UX-R91 Module Overview

The UX-R91 "expansion" module is fundamentally different from all of the other IC-900/901 modules. Even though it is receive-only, the PLL architecture is much more complicated due to the wide bandwidth it must cover (there are different combinations of three VCOs and two doublers needed to cover all of the frequency ranges of the UX-R91). The Front Unit for this module has a total of four OPC-179 cable connections to accommodate the additional interface signals required to facilitate the features offered by the UX-R91 (as is the case for the UX-S92). While the basic serial transfers are similar to those of the FM-only modules, there are some important differences, as discussed below.

The UX-R91 covers six frequency bands: the AM broadcast band (0.5 to 1.6 MHz), the FM broadcast band (WBFM, 76 – 108 MHz), the aircraft band (AM, 108 – 137 MHz), the 2M band (NBFM, 137 – 200 MHz), the 220 band (NBFM, 200 – 236 MHz), the 440 band (NBFM, 300 – 500 MHz), and the 800 band (NBFM, 800 – 950 MHz). The following table lists all of the controllable circuits that make up the UX-R91:

Control Description	Data Format (see text)	Strobe
Control data (CDAT)	8 bits, MSB first	CTRST
PLL (coarse frequency control)	60 bits, LSB first	PLST (CDAT[1] = 0)
VOL (stereo output volume)	18 bits, LSB first	PLST (CDAT[1] = 1)

The serial data protocols are described in Figure 17. Since the module cannot transmit, there is no provision for it to be connected as a main band, and therefore only sub-band interface signals are used. In addition to the power and antenna connections, the module features connections for right and left audio outputs from the WBFM demodulator. These outputs (as well as DETB, a mix of the right and left channel audio) are controlled by the volume (VOL) serial data register.



t1	clock half-period duration, 104µs (min)	t3	PLST setup/hold time, 10µs (min)
t2	strobe width, 10 µs (min)	t4	BAND6 hold time, 10µs (min)
D[mmmm]	PLL mode bits ("1000" typ.)	D[7:0]	Control data (see text)
D[N:a]	PLL data (56 bits, typ. "a" = lsb (first bit to shift)		

Figure 17. UX-R91 expansion module serial data diagram.

As shown above, PLST performs double duty acting as the strobe signal for one of two resources based on the status of a control data bit. This indicates that the control data transfers must be completed prior to loading these other resources. The control data registers produce logic signals that are used to control the module features and strobe steering.

© Joseph M. Haas, 2025, all rights reserved

As mentioned previously, the architecture of the UX-R91 PLL is quite different from any other UX module and features a reference oscillator input, multiple VCO sources, and several general-purpose outputs. The reference oscillator/divider establishes the frequency step (which is generally different for each band), while the GP outputs select the VCO/front-end path for a given band.

UX-R91 Control Registers

The UX-R91 features one 8-bit control register:

Bit#	CTRST (CDAT)
7	not used
6	mode (0 is FM, and 1 is AM)
5	not used
4	If 1, module power on, else off
3	not used
2	not used
1	If 1, select VOLST, else select PLLST
0	not used

UX-R91 Volume (WBFM Only)

The UX-R91 features a line-level, stereo output for the WBFM (76 – 108 MHz) band. A volume/balance control is implemented using a dual, digital level control I.C., the TC9154. This IC allows 35 distinct amplitude settings that are calibrated in dB. The attenuation step is 2dB and the range is 0dB to -68dB (which is essentially muted). The device features an 18-bit control register which holds all of the control bits. This register is loaded by setting CDAT[1] = 1, shifting the 18 control bits, then pulsing PLST high then low. Figure 18 details the bitmapped field formats for the volume device.

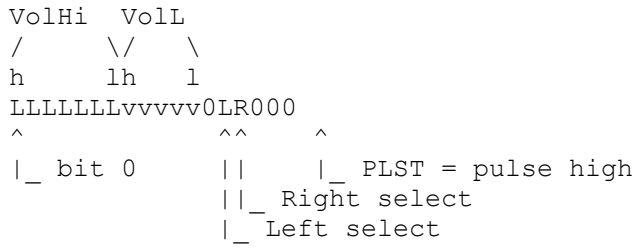


Figure 18. UX-R91 VOL frame field definitions.

The volume register implements the desired level setting as a “high” order and “low” order setting. The low order setting (denoted by the “L” bits in the diagrams that follow) selects 0 to -8 dB in 5 steps, while the high-order setting (denoted by the “H” bits) selects 0 to -60 dB in 7 steps. The two level values add to produce the overall level. The steps are implemented as a “1 of 7” binary code for the high-order value, and a “1 of 5” binary code for the low-order value. For example, the bit-pattern for a few level settings are as follows:

0 dB, Right:	-24 dB Right:	-68 dB Right:
HHHHHHHLLLLL0LR000	HHHHHHHLLLLL0LR000	HHHHHHHLLLLL0LR000
100000010000001000	001000000100001000	000000100001001000

© Joseph M. Haas, 2025, all rights reserved

Balance is implemented by applying a differential level setting to the right and left channels. The IC-901 implements these controls as step increments or decrements based on a button press on the control unit. Thus, there aren't really any calculations to make, just increment/decrement one or the other (for balance) or increment/decrement both together (volume). If one implements level and balance registers to control this feature, then it would be necessary to convert those MCU registers to the "1 of N" code format for each of the channels.

The UX-R91 provides a mix of the right and left channel output jacks for the DETB audio that is available at the Front Unit connector, so the volume and balance will affect this audio source even if the jacks are not used (DETB is, of course, a monaural mix of the right and left channels).

UX-R91 Operating Frequency

Even though the UX-R91 PLL/VCO sub-system is rather complicated, the frequency controls are relatively straightforward. All of the possible PLL configurations are transferred in a single 60 bit frame (even though the LM7005 PLL chip supports other frame lengths, this is the only one used for this application). There are a lot of bit-fields in the 60-bit frame, but they are reasonably straightforward to assemble.

Because of the multiple bands supported by this module, the PLL equation is best handled by sorting the calculation into one of 7 cases, based on the operating frequency (F_{op}). The other fields that are dependent on F_{op} are the R_{div} register, and the GP output register. Finally, the CDAT register has the modulation mode bit and the VOL/PLL steering bit that must be configured.

The PLL update stream consists of a CDAT transfer followed by a PLL frame (as depicted in Figure 17). The bit field diagram of Figure 19 illustrates the fields that are transferred with the PLL frame.

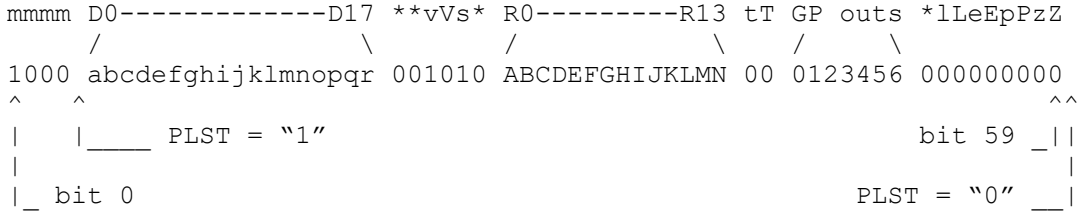


Figure 19. UX-R91 PLL frame field definitions. Gaps between bit-fields are added for clarity.

Note that the LM7005 requires that the shift order be lsb first, so the bits illustrated in Figure 19 would be shifted first from the left side of the figure (bit 0 first), progressing to the right (bit 59 last). The D[r:a] bits are the PLL N-divisor value (r = msb, a = lsb), the D[N:A] bits are the reference divisor value (N = msb, A = lsb), and the D[6:0] bits are the GP outputs (which are used to switch between the different VCO and front end circuits). All other bits hold the values shown for all frequencies.

The PLL calculation is similar to the FM-only modules except that there are 5 different equations that are utilized based on the desired F_{op}. Furthermore, the GP Outputs and reference divider value are other variables which all fold into the 7 different combinations needed to control the UX-R91 across its design frequency ranges. The PLL-N value (D[r:a]) equation follows the form of:

$$D[r:a] = ((F_{op} - F_{base})/F_{step}) + PLL_{BASE}$$

© Joseph M. Haas, 2025, all rights reserved

Where F_{base}, F_{step}, and PLL_{base} are determined by the value of F_{op}.

The following table lists the equation parameters for the seven band ranges of the UX-R91:

Freq. Range & Mod.		F_{base} (MHz)	F_{step} (KHz)	(18 bits) PLL_{base} (hex)	(14 bits) Ref (D[N:A])	(7 bits) GP out (hex)	Modulation CDAT[6]
0.5 – 1.7	AM	0.500	1.00	0x003bb	0x1900	0x7e	1
76 – 107.9	WBFM	76.000	5.00	0x03304	0x0500	0x7d	0
108 – 136.9	AM	108.000	12.50	0x0345c	0x0400	0x7b	1
137 – 199.995	NBFM	137.000	5.00	0x03c82	0x0500	0x7b	0
200 – 236.0	NBFM	137.000	5.00	0x03c82	0x0500	0x77	0
300 – 500.0	NBFM	137.000	5.00	0x03c82	0x0500	0x6f	0
800 – 950.0	NBFM	800.000	12.50	0x0e764	0x0400	0x5f	0

Note: CDAT[4] = 1 (module on) and CDAT[1] = 0 for the PLL frame (all other CDAT bits = 0)

As a software note, perhaps one of the easiest ways to handle the PLL frame is to construct a serial driver function that shifts a long-integer (32 bits) by a variable number of bits (up to 18) via the data and clock pins of the processor. This one function could then be used to send each of the fields defined in Figure 19 making it a relatively simple task to make sure that the bit alignment and bit count are correct.

© Joseph M. Haas, 2025, all rights reserved

IC-901 Control Frames

Up to this point, the descriptions of the module control data have focused on the data stream as it is presented to the module in question with the source being the ICOM controller or one of custom design. The origin stream produced by the IC-901 control head ("Remote Controller" in ICOM nomenclature) features additional bit fields that are used by the IC-901 base unit to produce the strobes and delays needed by the modules and various peripherals. While the module-oriented view can be used to control the external modules, the IC-901 Remote controller formats must be used to control the VHF and UHF modules which reside in the IC-901 Base Unit.

The control connection between the control head and the base unit is via a 2-wire, synchronous data connection that features clock and data signals, refered to here as the SYD stream. The clock rate is 4800 Hz which produces the corresponding baud rate. The transfers are sent as one or more 40 bit frames (the UX-R91 features a 60-bit control frame) that feature a single start bit (logic "0") and 5 or more stop bits (logic "1").

In addition to the SYD data stream, there is a 1-wire asynchronous data stream (herein referred to as the ASD) that the base unit sends to the control head to communicate various status conditions. This stream is a 4800 baud, asynchronous word that is 19 bits (including start bit) plus at least one stop bit (generally, there are over 35 stop bits between these status words).

Controller to Base Frames (SYD)

The following table identifies the primary "SYD" bit fields:

```
      msb                                  lsb
      |        1         2         3       4 _____
      12345678901234567890123456789012345678901 __/
SYD:  SgggguuuP.....................Pzzzzz
```

Bit count, first bit transferred is shown on the the left. This document aligns the msb as the first bit sent for both SYD and ASD.

```
      S = start bit (always "0")
      gggg = global addr ______ "peripheral address"
      uuu = UX addr _______/ (also control data for some frames)
      P..P = payload (number of bits and alignment depends on the peripheral address)
      z..z = stop bits (5 or more, always "1" -- data idles at "1" between frames)
```

All controller "messages" are configured as a 40-bit word (except for the UX-R91, which uses 60-bits). For the EX-766 optical cable interface, the clock signal between the control head and the optical A-Unit is reconfigured to be the PTT signal input to the controller IC. This means that the data stream output from the controller reverts to an asynchronous transfer (hence the presence of the start and stop bits). No other changes to the data field timing or format occur when switching between "Normal" and "EX-766". This configuration change is managed by S3 in the control head (select "EX766" or "Normal"). This switch is a SPDT type that simply selects the clock output from the controller IC ("Normal") or the PTT input to the controller IC ("EX-766) be connected to J1-5 on the controller. The controller IC itself doesn't otherwise know any difference between these two modes – it always looks for PTT on pin 44 and always sends the SYD clock on pin 43.

It should be noted that the IC-901 base unit SYD data transfer is clock controlled which means that one can, technically speaking, insert delays between bits in a frame and generally not see any deleterious

© Joseph M. Haas, 2025, all rights reserved

behavior as a result. However, the EX-766 requires that the SYD data stream be asynchronous (since the clock is re-tasked as the PTT input to the controller) so there can be NO delays between bits in the SYD control frames in this configuration. If one intends to ONLY control the IC-901 base unit directly, then it is not critical to maintain bit timings for the entire 40-bit (or 60-bit in the case of the UX-R91) frames. However, if it is desired to use the EX-766, one must ensure that there are no extraneous delays introduced into the synchronous data stream. Furthermore, the EX-766 use-case will also require that the 4800 baud timing be strictly maintained as well.

The controller identifies the target peripheral using the "gggguuu" bits in the frame (the "peripheral address"). The following list describes the available peripherals (the "S" bit is included to aid in aligning the bits to a hex representation):

```
Sgggguuu:
00001001 =    UX-19
00010010 =    UX-59
00011011 =    <unknown>
00100100 =    UX-39
00110110 =    UX-129
00111000 =    2m (base unit)
01000000 =    440 (base unit)
01001000 =    <unknown>
01010000 =    UX-S92
01011000 =    <unknown>
01100000 =    UX-R91
0111qr00xy =  TONE ------ "qr" = opt1/2 xy = ???
01110o01 =    SQU – "o" = opt
01110o10 =    VOL
01110011 =    <base reset frame>
00000000 =    <reset frame>
```

Power on reset depends on a specific choreography between the control head and base unit. First, the control head is powered on and sends a series of "reset frames" which are designed to assert "power on" in the base unit and place all supported modules into a known, power-idle state, to query those modules and note if they are detected. The completion of this sequence initiates the ASD words to begin (ASD data starts near the middle of the UX-R91 frame) starting with the ASD-INIT frame (which identifies which modules are installed). The following words (listed in HEX) are those sent during power-on:

```
0x000000001F              Base reset trigger
0x730000001F              Base peripheral clear
0x090000001F              UX-19
0x120000001F              UX-59
0x1B0000001F              <UX-29, not accessed by the IC-901 controller>
0x240000001F              UX-39
0x360000001F              UX-129
0x380000001F              2m (base)
0x400000001F              440 (base)
0x480000001F              <unknown, possibly 430 SSB>
0x500000001F              UX-S92
0x580000001F              <unknown, possibly 1200 SSB>
0x600000000000000007F     UX-R91
```

This list is static, it is sent as shown above at each power-on event.

© Joseph M. Haas, 2025, all rights reserved

Immediately following the initialization sequence, the control head sends the control frames to set the PLL, VOL, SQU, and TONE peripherals for the modules that are indicated for “main” and “sub” (the previous settings for main & sub are retained in battery-backed RAM located in the remote controller). At this point the IC-901 is a “radio” that is ready to receive signals. Transmitting signals depends on detecting PTT switch closure and sending the PLL frames needed to put the radio into TX mode. Also, tone decode and “pocket beep” operations require detection and response from the control head (more on these later). The following lists an example control frame initialization:

```
0x4080000A3F                    SUB = 447.375, TENC 88.5
0x4080000A3F
0x40804AA2DF
0x408000011F
0x408000007F
0x70C000181F                    TONE
0x710040811F                    SQU
0x720002111F                    VOL
0x39C002803F                    MAIN = 145.45
0x39C002803F
0x39C0190C9F
0x7080001FFF                    TONE
0x710040221F                    SQU
0x7200080A1F                    VOL
```

After the initialization frames, the radio is ready to operate and is in RX mode. Control of the radio basically follows a few distinct use-cases: 1) There is a PLL change, either due to an operating frequency change or a transition between RX and TX. 2) There is a change to the squelch or volume settings. 3) there is a tone setting change. After reset and initialization, most all of the SYD data frame sequences are triggered by one of those three scenarios. The General sequence for PLL changes is to send all of the PLL control frames for the affected main or sub unit, followed by the current tone frame for that unit. Volume or squelch changes require only the frames relating to the changed values. Tone frames alone are sent if there is only a change to the tone settings. The following sections describe the frame formats for each control type.

Note: The IC-901 system places the remote controller at the top of the “pyramid”. Generally, no control peripheral is changed unless the controller directs the change. The base unit does not perform any autonomous activity other than reporting status changes on the ASD port. Thus, without the controller, nothing substantive can happen in the IC-901 system.

Base Unit Frame

There are several base unit peripherals that are covered under the base unit frame address. The overall bit frame is mapped as follows:

```
         1         2         3         4
1234567890123456789012345678901234567890
SgggguuuuuuuuP...................Pzzzzz
gggg = address
u..u = control data
P..P = payload data
```

The next few sections deal with the VOL/SQU controls and various tone options used on the IC-901.

© Joseph M. Haas, 2025, all rights reserved

VOL/SQU Frame

There are two (2) volume controls, one for main and one for SUB and also two squelch controls (main/sub). The digital level control IC used for these levels is the TC9154 (the same as used in the UX-R91 volume controls) and thus the basic payload calculations are the same (instead of left/right, here the control select is main/sub – see "UX-R91 Volume (WBFM Only)" section for bit calculation details). The overall bit frame is mapped as follows:

```
         1         2         3         4
12345678901234567890123456789012345678901
SgggguuuP.........................Pzzzzz
01110oVQThABR0000HHHHHHHLLLLL0ms00011111
o = OPT select (if 1, OPT1=main, OPT2=sub, else OPT1=sub, opt2=main)
       If no UT-40 is installed, o = "1"
       This bit is present in the SQU/VOL/TONE Frames (NOT the UT-48 frames)
V = VOL _________ !!! Cannot have more than one = "1" !!!
Q = SQU ______/
T = Tone ___/
h = "1/2" = "1" if tone data goes to OPT1, else OPT2 if "0"
A = AFM1 = AF mute main if "1"
B = AFM2 = AF mute sub if "1"
R = repeat mode active
H..H = major step (1 of 7 = on)
L..L = minor step (1 of 5 = on)
m = main -----\___ !!! Cannot have both = 1 !!!
s = sub _______/
```

TONE Encode Frame (IC-901 Tone Unit)

There are several options for tone on the IC-901. First, there are encode and decode options (the encoder was generally included standard). The pocket beep option is another. The following describes the TONE encode frame:

```
         1         2         3         4
12345678901234567890123456789012345678901
SggggouuP.........................Pzzzzz
01110oVQThABR00000000000000cctttttt11111
V = Q = "0", T = "1"
cc = "11" (tone off or RX) or "00" (TX & tone enabled)
t..t = 6-bit tone code (0 – 38 decimal, follows "ENC code" bitmap in Table 4)
```

The encoder only applies to the main band. If the UT-40 is installed, the encoder is used if the UT-40 is assigned to the sub band. If the UT-40 is assigned to the main band, then the UT-40 is used for both encode and decode. *Note that the 6-bit tone code is different for the UT-40 vs. the encode module – see table below.*

Note: It is unclear why the "cc" bits are both active in the data traces. The schematic for the Tone Unit (IC-901 svc. man.) indicates that only the msb is connected (stream bit# 28). Activating both together is not going to cause issues with the tone unit, so it is advisable to follow the data trace result).

© Joseph M. Haas, 2025, all rights reserved

TONE Decode Frame (UT-40 Option Module)

The CTCSS decode option also allows the pocket beep option. Both are configured in the same fashion but are handled differently by the remote controller. As mentioned, the UT-40 must be assigned to the band (Main or Sub) where the decode operation is desired. Further, if the UT-40 is assigned to the Main band, it will be used for both encode and decode. Figure 18 is an annotated schematic of the UT-40 included for reference.

```
          1         2         3         4
12345678901234567890123456789012345678901
SggggouuP.........................Pzzzzz
01110o00100000000000000000cettttttt11111      decode
ce = “10” (RX) or “00” (TX)
ce = “11” to disable UT-40.  ce = “01” is not allowed
t..t = 6-bit tone code (0x15 – 0x39, follows the “DEC code” bitmap in Table 4)
```

Tone	ENC code	DEC code	Tone	ENC code	DEC code	Tone	ENC code	DEC code
67.0	0x01	0x1D	107.2	0x0E	0x36	167.9	0x1B	0x29
71.9	0x02	0x1C	110.9	0x0F	0x35	173.8	0x1C	0x28
74.4	0x03	0x1B	114.8	0x10	0x34	179.9	0x1D	0x27
77.0	0x04	0x1A	118.8	0x11	0x33	186.2	0x1E	0x26
79.7	0x05	0x19	123.0	0x12	0x32	192.8	0x1F	0x25
82.5	0x06	0x18	127.3	0x13	0x31	203.5	0x20	0x24
85.4	0x07	0x17	131.8	0x14	0x30	210.7	0x21	0x23
88.5	0x08	0x16	136.5	0x15	0x2F	218.1	0x22	0x22
91.5	0x09	0x15	141.3	0x16	0x2E	225.7	0x23	0x21
94.8	0x0A	0x39	146.2	0x17	0x2D	233.6	0x24	0x20
97.4	0x0B	n/a	151.4	0x18	0x2C	241.8	0x25	0x1F
100.0	0x0C	0x38	156.7	0x19	0x2B	250.3	0x26	0x1E
103.5	0x0D	0x37	162.2	0x1A	0x2A			

Table 4. CTCSS ENC/DEC Tone codes.

The table above describes the tone codes (hex) for each of the supported CTCSS tones. *Note: the 97.4 Hz tone that is available on the encoder is not used in the IC-901 because it wasn't supported by the UT-40 decoder (MN6520)*. The codes (hex) from the table are aligned with the “t..t” subframe and the “c” or “cc” bits set according to the tone mode.

Processing COS & TSQU Signals

The CTCSS decoder processing is mostly handled by the base unit (the rare exception to the non-autonomous rule described earlier). However, the busy LEDs (or other indicator of choice) for main and sub are managed by the remote controller. The controller must monitor the sporadic and prime ASD status words to determine when to illuminate channel busy for each of the active bands.

Basically, the TSQA status bit is ANDed with the COSA status bit to produce the main indicator (“1” = channel active/squelch open) and the TSQB bit is ANDed with the COSS bit to get the sub indicator.

© Joseph M. Haas, 2025, all rights reserved

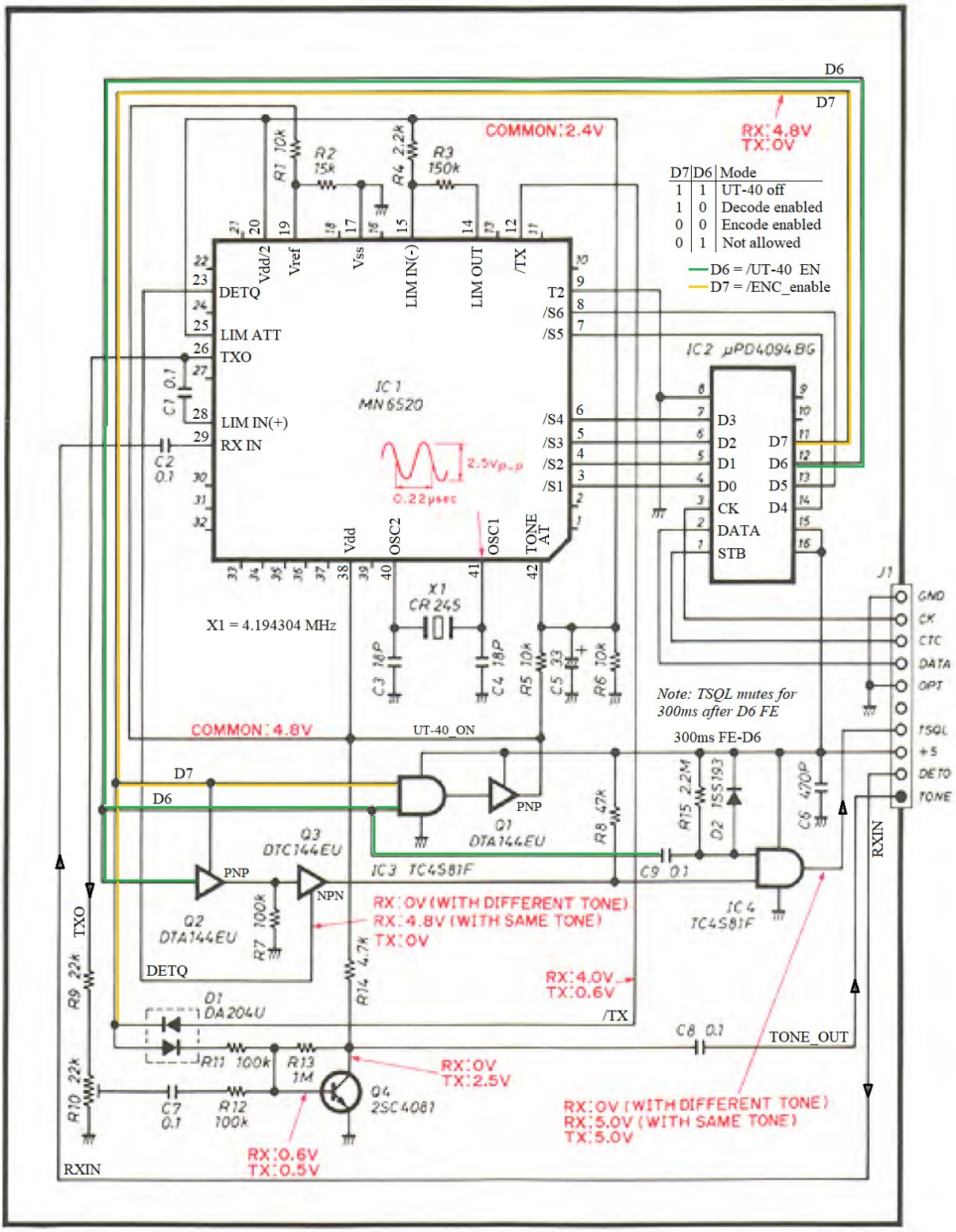


Figure 18. UT-40 Schematic (annotated).

© Joseph M. Haas, 2025, all rights reserved

DTMF Code SQU Frames (UT-48 Option Module)

```
               1         2         3         4
      12345678901234567890123456789012345678901
      Sgggguuu---P.....................Pzzzzz
      01111PQDM00aaaabbbbccccddddeeeeffff11111
      P = PAGE mode if “1”
      Q = Code squelch if “1”
      D = DATA if “1”
      M = AF Mute if “1”
      P..P = all “1”s unless data present
 SQU:  a..c = station code, 3, 4-bit DTMF code digits (0d – 9d)
 SQU:  d..f = all “1”
PAGE:  a..c = station code, 4-bit DTMF code digits (0d – 9d)
PAGE:  d..f = call-from code, 4-bit DTMF code digits (0d – 9d)
```

The UT-48 is a DTMF encode/decode option for the IC-901. The 901 uses this for code squelch (*Note: this protocol uses DTMF tones, NOT sub-audible continuous DCS*). The frame formats are just the basic framework of the system. There is higher-level protocol that must be implemented to orchestrate the code reception/recognition and transmission. The remote controller must interpret inputs and send control frames to generate outputs. This adds a layer of complication that extends beyond the basic control frame bit-maps.

Note: In the IC-901 controller, code memory “0” is the station code for the radio. The code in this memory is what is sent to the UT-48 when selecting “P” or “c” with the “MODE” button and is the code sent first in the paging mode.

Code squelch works as follows: The sending station transmits a 3 digit DTMF code. The UT-48 receives this code and sends a DTMF status frame (status ID = “11”) which contains the code. If the code matches the station code (DTMF code entry 0 in the IC901 controller) or any of the enabled codes in the other code memories, the controller sends SYD frames to open the squelch (e.g., 0x7A7FFFFFFF). The controller keeps the AFM bit low for 2 sec following the loss of signal (COSx) after which an SYD message sets AFM (e.g., 0x7AFFFFFFFF).

For TX, the UT-48 is pre-loaded with the station code when code squelch is selected by the operator (e.g., 0x7B8667FFFF). When the radio enters transmit, the PLL frames are bracketed by a frame that sets the DATA bit in the page message (e.g., 0x7AFFFFFFFF) before the PLL frames, and then a frame that turns off the DATA bit (e.g., 0x7080001FFF) after the PLL frames. This triggers the UT-48 to send the 3-digit station code.

Note that the description above is for the IC-901. Other scenarios can be implemented so long as the basic steps listed above are followed.

Paging (Pocket Beep)

<TBD>

DTMF paging works as follows: The sending station transmits a 3-digit DTMF calling code, a DTMF “*” tone, then its 3-digit radio code. The UT-48 receives this code and sends a DTMF status frame

© Joseph M. Haas, 2025, all rights reserved

which contains the code. If the code matches the station code or any of the enabled codes in the code memories, the controller sends SYD frames to open the squelch.

Radio Module Control (PLL) Frames

As previously presented, each UX module (and the two base-unit radios) feature their own particular bit-mapped control formats. These formats simply need to be aligned into the IC-901 40-bit frame format to be sent to the base unit. The Base Unit modules (2m and 440) are discussed first.

IC-901 Base-Unit Module Control (PLL) Frames

The two "modules" contained in the IC-901 base unit are unique in that they are "always there" – they cannot be removed physically nor disconnected internally. The following describes the control formats for these radios. Note that both modules us the same PLL reference oscillator, 12.800 MHz.

```
              1         2         3         4
      1234567890123456789012345678901234567890
2m    00111abSPLBT000ppppppppppppppppppppp11111            PLL2001, Ref osc = 12.8 MHz
440   01000abSPLBT000ppppppppppppppppppppp11111            TC9181 Ref osc = 12.8 MHz
      ab = modulation mode ("00" for FM)
      S = "1" for SUB, "0" for MAIN
      P = unit power, "1" = on
      L = "1" for Low RF power, "0" = High
      B = Band, "0" is normal, "1" is out-of-band (affects IF filter BW)
      T = "1" for TX, "0" for TX off (RX)
      p..p = PLL data payload (align calculated data to right-most "p" position)
```

2m PLL (PLL2001)
The RANGE input (pin 14 of the PLL chip) is tied high in the IC901. This allows VCO inputs from 5 to 250 MHz at pin 8. This also allows all 16 bits of the input register to be used. The right-most bit of the register data is the CONTROL bit which selects the REF divider setting if "1". The ICOM protocol is to set the reference each time the PLL divider is changed. This is likely a fail-safe to ensure that the reference setting is correct in case it was corrupted. The reference setting is 12.8 Mhz/5KHz = 2560d (0xA00) left shifted 1 bit and a "1" inserted at bit0. This gives 0x1401 as the p..p data for the reference initialization.

The reference frame is followed by the PLL divider data. The divider setting is from the following formula:

div = (Fop – 136 MHz)/5KHz + 0x05CD0 + (tx * 0x0D70)
div = (Fop – 136 MHz)/5KHz + 23760 + (tx * 3440) {decimal operands}
Where Fop is the operating frequency, and tx = "0" for RX and "1" for TX

The result of the above calculation is then left-shifted one bit (with a "0" in the right-most bit position) to align with the "p..p" sub-frame.

A complete 2m frequency setting thus will consist of a reference init frame and a PLL divider frame. The following example is for 145.45 MHz, TX (low power):

```
0x39D002803F
0x39D01C689F
```

© Joseph M. Haas, 2025, all rights reserved

440 PLL (TC9181P)
The TC9181 is a much more complicated PLL than the PLL2001. In addition to the reference initialization and PLL divider, there are settings for lock detect, phase detect, and GPIO signals. Each of these play a role in the message frame construction. Consequently, there are generally 4 frames sent for each PLL update for the UHF band on the IC-901. Further complicating matters is the fact that the TC9181 requires data to be shifted LSB first, rather than MSB first as with most SYD peripherals. The data herein is calculated "normally" (msb on the left) with the expectation that the bit reversal will be handled before the data are inserted into the 40-bit frame.

The two most significant bits of the TC9181 word can be treated as "address" bits which select the type of control data being sent to the device. "00b" is the HL config, "01b" is the Fdiv data, "10b" is the REF osc divide setting, and "11b" is the GPIO setting. For every frequency change, all 4 control frames are sent in the following order: REF, PLLDIV, HL, GPIO.

440 REF Frame [10]
As with most of the modules, the REF divide value sets the Fstep of the synthesizer which is fixed at 5KHz for the IC-90x modules (except for the UX-129 which uses a 10KHz Fstep). The same reference oscillator is used for both 2m and 440 which results in the same divider setting. The register address of "10b" is placed at the msb bit positions of a 16 bit variable and the L2/L1 bits (fixed at "00b") at bit positions 13 & 12. The reference divider value, 0xA00, occupies the bit positions 11 through 0. The entire frame pattern is bit reversed and right aligned into the "p..p" field of the 440 control frame.

440 PLL Divisor Frame [01]
The calculation of the 440 PLL divisor is similar to that of the 2m calculation:

div = (Fop – 400 MHz)/5KHz + 0x12061 + (tx * 0x181f)
div = (Fop – 400 MHz)/5KHz + 73825 + (tx * 6175) {decimal operands}
Where Fop is the operating frequency, and tx = "0" for RX and "1" for TX

Once the divider setting is calculated, a "0" bit must be inserted (bit stuffing) after bit 5 (with bit0 the lsb) to account for the prescaler used in the synthesizer (just as with some of the IC-900 UX modules). One way of accomplishing this is illustrated in the following C code:

```
U32 bit_stuff(U32 pd){
U32  i;  // temp
#define LOW_MASK 0x3F     // mask lower 6 bits to insert 0 at b6

     i = (pd & ~LOW_MASK) << 1;
     i |= pd & LOW_MASK);
     return i;
}
```

Lastly, the frame must be bit-reversed within an 18 bit field, and inserted into the control frame right-aligned in the "p..p" field. The following is an example of a C function to perform an 18-bit reverse:

```
U32 bit_rev(U32 pd){
U32  i;  // temps
U32  j;
U32  r = 0;

     // Change the "i" assignment in the for loop to select the limit of bits to shift
     for(i=0x20000, j=0x00001; i; i>>=1, j<<=1){
          if(pd & i) r |= j;
     }
     return r;
}
```

© Joseph M. Haas, 2025, all rights reserved

440 HL Frame [00]
The HL frame controls the lock detect and phase detector output formats. It is fixed at “00b” for the IC-901. Thus, the bit-reversed control data right-aligned to the “p..p” field is 0x0.

440 GPIO Frame [11]
Normal operation in the USA forces the filter control bit to “0”. Thus the GPIO “p..p” frame, bit reversed and right aligned, is 0x3. To change the filter setting, the “p..p” data is 0xb.

The following is an example of a 447.375 TX, low power control sequence:

```
0x4190000A3F          { REF config }
0x419181BADF          { PLL divisor config }
0x419000001F          { HL config }
0x419000007F          { GPIO config }
```

IC-900 UX Module Control (PLL) Frames

The IC-900 featured 6 RF radio modules. Of these only the following are supported by the IC-901: UX-19 (10m), UX-59 (6m), UX-39 (220), and UX-129 (1296). The IC-900 module control frames use the same 40-bit control frame as for the 2m & 440 base modules. The following lists these modules 40-bit control frames:

```
              1         2         3         4
      1234567890123456789012345678901234567890
10m   00001001MSPLBT0ppppppppppppppppppppp11111     PLL2001   Ref osc = 12.25 MHz
6m †  00010010MSPLBT0ppppppppppppppppppppp11111     PLL2001   Ref osc = 12.25 MHz
220   00100100MSPLBT0ppppppppppppppppppppp11111     MB87001   Ref osc = 5.12 MHz
1296  00110110MSPLBT0ppppppppppppppppppppp11111     TC9181    Ref osc = 6.4 MHz
      M = “0” for SUB, “1” for MAIN---\___ “10” or “01” are the only allowed values !!!
      S = “1” for SUB, “0” for MAIN __/
      P = unit power, “1” = on
      L = “1” for Low RF power, “0” = High
      B = Band select (RX IF), “0” = in-ham-band, else reduced IF filtering
            † - Band select available.  Otherwise, “B” has no effect
      T = “1” for TX, “0” for TX off (RX)
      p..p = PLL data payload (align calculated data to right-most “p” position)
```

For the 10m and 6m frames, the right-most p..p bit is the register select which is “1” for REF frames, and “0” for Frequency divider frames. For 220 frames, the PLL divisor must be bit-stuffed (after bit5). Also, note that the 220 PLL chip uses a hard-coded reference divisor (1024) so there is no SYD REF divisor control frame for this module.

The UX-129 is configured in the same manner as the base-UHF module. However, the GPIO bits are more complicated in that they drive the XIT/RIT feature of the module. The 4 possible GPIO patterns correlate to the following functions: “00b” XIT/RIT reset, “11” XIT/RIT idle, “10” XIT/RIT up step, “01” XIT/RIT down step. To accomplish a step, a 40 bit frame is sent with the “up” or “down” pattern, followed by another 40-bit frame with the “idle” pattern. Send the “reset” followed by “idle” bit patterns to clear the count to 0 (this is the center of the tuning range). The tuning can be adjusted “up” 7-steps, or “down” 8-steps.

© Joseph M. Haas, 2025, all rights reserved

The following list depicts some examples of the 10m, 6m, 220, and 1296 control sequences:

28.000 MHz, Main RX
```
      0x09B00264BF                    REF divisor
      0x09B0078EDF                    PLL divisor
```

52.525 MHz, Main RX
```
      0x12A00264BF                    REF divisor
      0x12A00CFDDF                    PLL divisor
```

223.500 MHz, Main RX
```
      0x24B028459F                    PLL divisor
```

1270.000 MHz, Main RX
```
      0x36B000143F                    REF divisor
      0x36B041DDDF                    PLL divisor
      0x36B000001F                    XIT reset active
      0x36B00001FF                    XIT idle (reset released)
```

IC-901 UX-R91 & UX-S92 Module Control (PLL) Frames

<TBD>

Base to Controller Status Frames (ASD)

The following format describes the ASD status word:

```
               1         2
      12345678901234567890..n
ASD:  SiiP..............Pz..z
      S = start bit (always “0”)
      ii = status ID (“00” thru “11”)
      P..P = payload (number of bits depends on the status ID)
      z..z = stop bits (1 or more, always “1”)
```

There are four (4) types of status frames which are sent to the control head (the following have the start bit removed for clarity): *Note: the start bit is removed in the following ASD frame maps.*

```
00: Initialization status (reports the modules detected during power-on)
              1         2
      2345678901234567890
      00abxc0dq24rsw1111...
      a = UX-19
      b = UX-59
      x = <unknown>
      c = UX-39
      0 = <unknown>
      d = UX-129
      q = 1200 SSB (proposed)
      2 = 2m (base)
      4 = 440 (base)
      r = UX-R91
      s = UX-S92
      w = 430 SSB (proposed)
```

© Joseph M. Haas, 2025, all rights reserved

This frame is sent only once after power up and after the startup reset sequence is sent by the control head. It simply identifies all of the units installed. Clearly, there were plans to provide additional modules but these plans were abandoned by ICOM.

```
01: Prime status (sent every 7 ms)
              1         2
       2345678901234567890
       01bmsSSSSssss11111...
       b = busy (not used)
       m = main squelch open if “1” (COSM)
       s = sub squelch open if “1” (COSS)
       SSSS = main SRF value (4-bit, unsigned value)
       ssss = sub SRF value (4-bit, unsigned value)

10: Sporadic status (sent every 110 ms, or sooner if a PTT or other change)
             1         2
     2345678901234567890
     10PABudsTabc111111...
     P = PTT           “1” if PTT pressed
     A = TSQ A         main T.SQ detect
     B = TSQ B         sub T.SQ detect
     u = mic UP        “1” if “UP” pressed”
     d = mic DN        “1” if “DN” pressed”
     s = SCAN          u/d held for more than 500 ms
     T = Tone          CTCSS unit connected to OPT1
     a = OPT 1         T.SQ unit connected to OPT1
     b = OPT 2         T.SQ unit connected to OPT2
     c = OPT 3         DTMF unit connected to OPT3

11: DTMF Page (SQU & “pocket beep”) status (sent only if there is a change)
              1         2
       2345678901234567890
       110Addddddddddddd11...
       A = Code id (0 = RX to be sent, 1 = calling code to be sent)
       d..d = code, msb left (3-digit hex, nybbles are DTMF codes)
```

The pager function is active: All bits high if no code received.

Deep Thoughts on Control Frames

The information presented herein has largely been compiled using trace data from live IC-901 radios and schematic analysis. What is not accessible in these endeavors is the internal logic of the base-unit MCU. Many of its behaviors can be inferred based on observed inputs/outputs, but this only covers the input cases that have been presented and observed. Other input cases may be interpreted differently with outputs that differ substantially from the observations and inferences that were observed in preparation of this document. This section offers some information on timing observations. The recommendation is to preserve these timing values for the simple reason that they are known to work.

Baud Rates
4800 baud is the rate used by the IC-901 for both the status and control frames. This equates to 208.3µs per bit. For the synchronous clock, the clock edges should be 104.2µs apart and as close as possible to 50% duty cycle.

© Joseph M. Haas, 2025, all rights reserved

SYD Frame Delay

The delay between the end of one 40-bit frame and the start of the next should be managed so that at least 1.4ms separates the last stop-bit (the 40th bit in the frame) and the start-bit of the next frame. The delay between the end of the reset sequence and the first initialization frame should be at least 6.3ms. These values are minimum observed readings taken from many timing traces of these events.

ASD Frame Delay

The delay between the end of one ASD frame and the start of the next is determined by the base unit processor. Generally, about 3.5 to 4 ms of delay is present between these frames. However, the frames are accelerated if there is a change detected in the up/dn buttons, OPT module status change, or a change in in the PTT. In these cases, as little as one stop-bit of delay may be all that separates the stop-bit/start-bit between frames.

© Joseph M. Haas, 2025, all rights reserved

Closing Remarks

This document is a work in progress. In this latest revision I have corrected several typos and have included detail that supports the IC-901 base unit resources. At a minimum, this document should now serve as a reasonably complete reference covering the basic operation of all of the IC-901 subsystems. I have made every effort to verify the integrity of the information presented, but it is inevitable that an error or two yet linger herein. As well, I am still developing applications for the UX modules and am finding that there are still details to learn, even though I have been studying them for many years. Point in fact, this revision corrects many "bugs" in the PLL calculations for the standard modules (the curse one suffers with lack of peer review).

© Joseph M. Haas, 2025, all rights reserved

References

Datasheet, PLL2001, Nippon Precision Circuits, Tokyo, Japan (no date)

Datasheet, MB87001A, Fujitsu Microelectronics, September 1995 (Edition 8.0a)

Datasheet, uPD2834C, NEC, dated 1980

Datasheet, TC9181P, Toshiba, dated May 19, 1999

Datasheet, LM7005, Sanyo document # 2810A

Datasheet, TC9154AP, Toshiba, no date

ICOM Service Manual, IC-900A/E

ICOM Instruction Manual, IC-900A/E

ICOM Service Manual, IC-901A/E

ICOM Instruction Manual, IC-901A/E

IC-901 CPU SERIAL DATA CONTROL, Author unknown, Publisher unknown, 01/23/1992 https://www.repeater-builder.com/icom/pdfs/ic-900-901-data-structure.pdf (as viewed 7/19/25)

ACC FC-900 Interface Schematic, 3/1/1989

RBI-1 Generic Data Manual, Doug Hall Electronics, Columbus, OH, 2/23/1995

QuickFilter QF1Da512 datasheet (Rev A18, 4/22/2010) and QFPro software (www.quickfiltertech.com)

https://ke0ff.github.io/RDU900/index.html *IC-900 Remote Controller Clone Project Page (in 3 parts)*, by Joseph Haas, KE0FF, last update 07/22/2022.

https://ke0ff.github.io/mfmic/IC901_MFMIC.pdf *Adding Multi-Function Mic Capability to an IC-901*, by Joseph Haas, KE0FF, 03/08/2024.

© Joseph M. Haas, 2025, all rights reserved

Revision History
Rev B, 07/27/2025
Added IC901 control data sections
Revised PLL calculation page updating UX-129 and IC-901 UHF data
Various typo corrections
Updated email address

Rev A, 9/22/2019
Prior Rev notes not available.

© Joseph M. Haas, 2025, all rights reserved