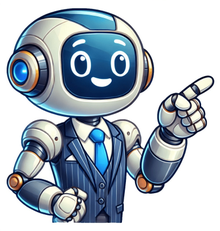
Continue



 Rev. 7/17/2021. Like most modern ham radios, the Icom IC-7100 comes with a built-in USB audio and serial devices, providing a one-cable connection to a PC without needing an external interface box, dongles, or audio cables. What this means is that the radio contains a USB hub chip, two serial port chips, and a soundcard chip (just as you'd find in external interfaces) that presents a soundcard and two serial (COM) ports to the computer. To use them you just set them up, then pick them from a list in your software's audio and CAT setup, just as if they were external USB dongles or computer-to-radio interface. In effect, they added a SignalLink USB-type device to the radio. But we still need to set up those devices, so follow along by opening and adjusting your Windows "Sound" dialog box, and the radio's Set → Connectors menu. Update (2021):See also this 2018 ICOM Knowlege Base PDF: Configuring Icom radios for USB Operation. Save the PDF, as it has much more info, with dialog box pictures, about setting up the computer, radio, and several popular digi-mode programs. Read that, then return here and read details for the 7100. Install the Serial Driver (Skip to Audio Setup, if you've already done this) Do NOT plug in the USB cable before downloading and installing the Icom USB Driver! If you do, Windows will automatically install a generic serial driver, and you'll have to go into Device Manager and remove it, then install the Icom-provided driver. Actually, it's not an Icom driver (though you get it from them), it's a Silicon Labs CP210x driver. That's the manufacturer of the USB-to-serial UART chips used in the radio. The soundcard chip in the 7100 requires no special driver; Windows idetifies it as a "USB Audio Codec" and uses a generic driver, just like if you plugged in any USB soundcard dongle. Read the Icom USB Driver Installation Guide. Save the PDF for future reference. It has detailed instructions and screen-shots for Windows 7, 8, and 10. Download the latest USB Driver (Version 1.30 {or latest}), Driver Utility and manuals. Unzip it and click CP210xVCPInstaller_x64.exe (or _x86 for 32-bit Windows). AFTER installing the driver, reboot, then plug in the USB cable, and then turn on the radio. In Device Manager, and programs that access COM ports, you'll find two new (virtual) COM (RS-232 Serial) ports, named like COM3 and COM4 (or 5&6, or 7&8, if you already have some real or virtual COM ports). They will appear as "Silicon Labs CP210x USB to UART Bridge" in Device Manager, under the "Ports (COM & LPT)" branch of the device tree. The first com port will be called USB1 in the radio's docs and menus. This is the CAT (Computer Aided Transceiver, Icom CI-V commands) port you'll use in all your radio software that provides and uses radio control. The second COM port is called USB2 in the radio. It can be set to output RTTY or DV data messages (to an ASCII terminal emulator software), or used for a NMEA GPS for D-Star. The radio's serial UART chips present RTS and DTR states, but the 7100 does not use them, so forget about PTT or CW keying via virtual hardware pins in your digi-mode software. You must use software that will PTT via CAT command, so older software that only sends PTT and CW keying via serial pins will not work. Update 1: I found a little device for $10 that simply plugs into a USB port and the radio's CW Key jack. It apparently contains (in the USB plug) a USB to UART bridge chip (it presents a virtual serial port to the computer), and uses RTS and a switching transistor to key (ground) the radio's CW jack. It works fine. It's called "LD-C103" and is a generic/unbranded part. Update 2 (12/02/20): fldigi now supports CW keying via CAT! So no external keying interface needed. Thanks, Dave! In your rig control software settings, pick the first COM port and set it to 19200, N, 8, 1, and CTS/RTS doesn't matter; I set them to "Forced Off." Note that if you have a 7300, you can use them and there are settings for their functions in the 7300's Connectors menu. Note also that two programs cannot share the COM port. If for example you have OmniRig running in the background, then start HRD, you'll get that good ol' DOS-days error, "COM port already in use."Actually, there's a way around that—a utility called VSPE (Virtual Serial Ports Emulator), whereby you map a real (or virtual) serial port to a new virtual port, then point as many programs as you like to the new port, and all can use it at the same time. Why? You might run HDSDR as a panadapter, with OmniRig providing CAT control to it, and you might also run your digi-mode software, and logging software, all of which need access to the radio's CAT data on the same COM port. Strangely, the 32-bit version is free, but the 64-bit version costs about $20. VPSE has a nice graphical interface and is easy to set up, then forget. Note that once you've set it up, you need to save it's setup, and tell it to start automatically with Windows. Then remember to set all your software to use VPSE's port, not the radio's port, for CAT. Note (12/20): VSPE cannot route or sequence CAT commands and replies (CAT is just serial hex strings, not packets with address headers). If you use two or more programs at the same time on VSPE, there will be CAT data collisions that will cause the radio and/or the software to do strange things. A partial solution is to reduce the polling times in the software to perhaps 2000 ms. Also, the JT code-based programs (WSJT-X, JS8Call, FSQCall) will not even start up if another program is on the VSPE port. One work-around that I've tried is using OmniRig for both programs. Set Rig1 and Rig2 to the same rigfile and port (the VSPE virtual port). Then set software 1 to use Rig1 and software 2 to use Rig2. This trick only works for programs that can use OmniRig. There's also a free one called com0com, but its setup is a bit too confusing, so I didn't mess with it. The Icom USB connection also presents the USB soundcard chip in the radio, and it's just like any USB Soundcard dongle or interface box—it shows "Speaker (USB Audio Codec)" and "Microphone (USB Audio Codec)" in the Sound dialog and anywhere you can select audio devices. Audio Setup Locate Windows' "Sound" dialog in Control Panel. Or, right-click the little speaker in the taskbar and choose Sounds. If you end up in the Settings app (in System, Sound), look to the right under Related Settings and click Sound Control Panel. You'll initially use the Sound dialog quite often, so it's a good idea to locate "Sound" in Control Panel and add an icon for it to your desktop, then pin it to the taskbar. In the Playback and Recording tabs of the Sound dialog, open Properties of both devices and rename them to something more meaningful: Playback → Speaker (USB Audio Codec) → Properties → General, rename it to IC-7100 XMIT Recording → Microphone (USB Audio Codec) → Properties → General, rename it to IC-7100 RECV And in both of them, click the Change Icon button and set it to the 2-jack RCA cable icon. Showing a microphone that gets the radio's output (like a speaker), and a speaker that modulates the radio (like a microphone), is rather confusing, as are the generic device names... Yep, in good ol' Windows, your radio transmits with a speaker and receives with a microphone. Why can't they just say Input and Output? So don't skip doing the above—it'll make everything easier, as the names you give the devices will appear in your digi-mode software configuration settings. Set Audio Levels Windows Sound Levels The USB audio levels will initially be set to 100% — EEK! Set them as shown below before starting any digi-mode software. There is a known bug in the TI sound chip used in many modern amateur radios. This chip mis-identifies itself to Windows as a microphone device, causing Windows to add 30 dB gain to it. The solution is to lower the level slider, almost to the bottom, to -0.4dB. This applies to Windows 7 in 2018. Update (2019): Windows 10 apparently identifies the radio's sound chip properly and applies the normal line-level gain range. The very low gain setting described herein applies to Windows 7, and perhaps 8. In the Recording tab, open properties for the IC-7100 RECV (Microphone (USB Audio Codec)) device, Levels tab, right-click the level slider and change it from percent to deciBels. To verify if your chip has the bug, lower the slider to minimum, then use the right-arrow key to raise it 3 clicks, til it shows -0.4 dB. But if your slider shows -96 to +30 dB, with 0dB in the middle, you don't have the bug and/or Windows has properly identified the chip as a line-in device, and a medium level (0dB or 50%) is where you initially set it. In the Playback tab, and open properties for the IC-7100 XMIT (Speaker (USB Audio Codec)) device. Set its level to -24 dB (20%). On my Windows 7 computer, I had to use 13%, but on the Windows 10 computer I use 20-25. YMMV. You will also set the radio's USB MOD Level to 20. The idea is to balance them, not have one very high or low, lest you overdrive the audio chip and create distortion. And both must be set so the radio isn't driven to any ALC indication, which is usually 25-30 watts on PSK-31, or 50-60 watts on a pure audio tone. The settings above may be wrong for you, because Windows keeps changing the way it handles audio devices. Just be aware that your Recording (receive) level may be normal, or may need to be set real low. You'll soon know, when you send received audio from the radio to the computer, watch the green bar-graph in the Sound dialog. It should be bouncing around well under half-scale on band noise, and about three-quarters on a strong signal. The loudest audio the radio can send is unsquelched FM mode hiss, which you should set to nearly at the top of the green bar-graph in the Sound dialog. Write down the settings. Note (2021): While in the Playback tab, make sure that the "Default" playback device (with the green check mark on it) is NOT set to the Icom XMIT (USB Audio Codec) device, but is set to your usual speakers or HDMI audio. Windows chimes and other sounds are sent to the "Default" device. If the radio is set to Default, it will transmit Windows sounds and whatever is going to your default sound playback. Monitoring If you want to listen to the radio through the computer, you need can go to Sound → Recording → IC-7100 RECV → Properties → Listen, and check the "Listen to this device" box. It should tee the audio to your default playback device, usually speakers or HDMI audio. It will have some latency (echo) unless you turn down the radio. For monitoring your transmitted digi-mode audio, there's no Listen option for playback devices. Turn on Set → Functions → Monitor in the radio. It's helpful to monitor the transmitted audio in digi-modes; to know what you're actually transmitting, when the tail stops, etc. Otherwise, you might not know you're still transmitting, if your software macros didn't go to receive as expected, and you're not looking at the watt meter. I never want the computer transmitting anything I can't hear. Radio Levels Next, in the 7100, press Set → Connectors: USB AF Level: 50 (default) USB MOD Level: 20 ← IMPORTANT: The input is WAY too hot at the default 50%! ACC MOD Level and DATA MOD Level: set to 0%, if you're not using those inputs. In this case, DATA MOD Level doesn't mean SSB-DATA mode modulation, it means the DATA jack on the back of the radio, where you'd connect something like a TNC. DATA OFF MOD: set to MIC, ACC: disables mic and accessory jack inputs in SSB-DATA mode; enables them when NOT in DATA mode. (DATA OFF MOD means "MODulation source when DATA mode is OFF", i.e., regular voice mic operation. DATA MOD: set to USB. ACC is the default, so it will not transmit USB audio until you set this! In this case, DATA MOD means "MODulation source when DATA mode is ON." So, DATA MOD Level and DATA MOD are two entirely different things. Confused yet? With DATA MOD set to USB, level is controlled by USB MOD Level, and DATA MOD Level is only for the radio's data DIN jack. They could have done that a little better in the firmware... On my Windows 7 system, with Playback at -30.9 dB (12%) and USB MOD Level at 21% produced the following results: In DM780: 29 watts on PSK idle, 63 on PSK-off tone In DM780: 50 watts on RTTY idle In MixW3: 31 watts on PSK idle, 65 on menu Commands → Tune (it outputs a single tone but may also be set to two-tone) In MixW3: 66 watts on RTTY idle On my Windows 10 system, Playback at 20%, radio USB MOD Level at 20% produces the correct 30 watts on PSK-31. So, if you're on Windows 10, start with 50 and 50 for receive, and 20 and 20 for transmit. But in Windows 7, the receive (mic) will be near minimum, and the transmit (speaker) will be 12-13. YMMV. 2021: They changed it again! Now, without changing the radio (20 and 50), I need to run 26 on XMIT and 70 on RECV, in the Sound dialog. I have not tested this combination for distortion, I just set proper power levels. Also note that the USB MOD Level control is very sensitive — 20 produced 30 watts PSK and no ALC, while 25 produced 117 watts and full ALC (the default 50% would likely destroy something). You never want ANY ALC on digital modes, as it degrades the quality and readability of the signal, and splatters all over everybody. More Notes: The 7100's default USB MOD level of 50% is WAY TOO HIGH (sensitive), requiring setting computer level to 0-1% (no granularity). Reduce USB MOD to 20% then set computer level to 12-26% to set RF power to about 30 watts (no ALC indication) on two-tone (or PSK), or about 60 watts single tone (or RTTY), from your digi-mode software. Do NOT limit RF power with the RF Power control, leave it at 100% (well, at least 60%) and limit power with audio drive level. In Icoms, turning down the "RF Power" control just raises the ALC bias level, and we want no ALC action of any kind. The same applies to ALC from an amp - leave it up at your normal tune-up power level. The idea is to have plenty of head-room in the linear range of both the radio and the amp (if used). Always decode and send digi-modes at 1500 Hz. Why? The 7100's filters center on 1500 in both SSB and SSB-DATA modes. So, if you tune somebody at 1000 on the waterfall, then dial the filter down to 100Hz, oops! Where'd he go? Outside the filter, which at 100Hz is from 1450-1550! Yes, you can move the filter's center by twisting both PBT knobs, but unfortunately it won't shift it past the filter's current bandwidth. Regarding #3 above, In DM780, set: Program Options → Waterfall → Center Frequency Marker → 1500. Clicking the C tag will tune the radio to put the selected signal at 1500. In fldigi, set Configure → Miscellaneous → Sweet Spot, set CW to your CW offset, as set in CW Pitch in the radio), and RTTY and PSK to 1500. Then clicking the QSY button below the waterfall will tune the radio to put the selected signal at 1500. Then you can click on a signal anywhere in the waterfall, then click the C tag (DM780) or QSY button (fldigi), and it will instantly tune the radio to move the selected waterfall signal to 1500 Hz. Then you can select narrow filter and start the QSO. It's like the CW Tune button on the radio, which tunes the signal to be at your selected CW offset. Also in fldigi, you can add macro commands that will QSY and set the filter when you answer a CQ, then set the filter back to wide when you 73 the QSO. Nifty! What's Next? Calibrating the IC-7100 73, — KV5R Page 2 © 2002-2011 by Harold Melton, KV5R. All Rights Reserved. A Popular Project Lots of people are building this antenna! One fellow even improved upon it by making the elements out of aluminum - inside a ¾″ PVC pipe — see QST August 06. Antennas just cost way too much! You can build this one for well under $10. It's a: cheap, easy, stealthy, good performing, no ground plane, portable/mobile/marine/base, fun project antenna! This is an off-center-fed sleeve dipole, made of ½″ CPVC and aluminum foil tape. The elements are fed 3-¼ inches below center, with the coax inside. Off-center feed is required because of the interaction of the lower element with the coax inside. Note: if you make it center-fed, the SWR will be about 2.5:1 across the band. Lowering the feedpoint by 3-¼ inches lowers the SWR to under 1.5:1. It is very broadband, being useable from about 142 to 152, and all of the 440 band as a 3/2 dipole. Also works quite well as a VHF/UHF public service band scanner antenna. Note: I do not claim origination of this general design. OFC sleeve dipoles have long been used as marine HF antennas, and more recently, are widely used in small UHF wireless consumer devices. This dipole was constructed as follows. Cut 7 feet of ½″ CPVC. Drill 7/32nds holes at 22-¼ and 58 inches from the "top" of the CPVC pipe. Cut upper element tape (2" wide aluminum duct tape) at 22-1/8th inches. Make two. Cut lower element tape at 15-5/8ths inches. Make two. Apply two overlapped layers of the tape to the upper and lower parts of the CPVC, leaving a ¼ inch gap at the hole which is 22-¼ from the top. The overall length of the two elements should be 38 inches. Take 6-12 feet (or as needed) of RG-58 and apply the appropriate connector to one end. Strip about an inch of the other end; fold the shield back. Bend a curl in the end of the center conductor. Push the coax into the lower hole with the curl "up" and guide it into the pipe, pushing and twisting as needed, until it pops out of the upper hole, between the elements. Strip out about 5/8ths inch of the center conductor. Apply a little Penetrox to both center and shield. Pull the coax back until just the conductors exit the hole. Lay the greased conductors on the elements and tape them down with a couple little strips of aluminum tape. Tape the whole feed with several layers of tightly-stretched electrical tape (Scotch 33+). Secure the end of the tape with small cable ties. Seal the ends to keep out water and bugs. To make it look cool, spray paint it glossy white, dry, then spray every other six inches with flourescent orange (portable or bicycle); or olive drab ("covert" ops). I painted mine white because that's what I had handy. Don't use conductive (metallic) paint! The same idea (CPVC and foil tape) may be employed to build small yagi antennas also. Ignore the ruler — it should read 22-¼ (from the top), not 19 (photos taken on first try). The coax conductors are connected to the tape elements by being (1) greased, (2) sandwiched between aluminum tape, and (3) compressed with several layers of tightly-stretched electrical tape. Again, ignore the ruler — it should read 22-¼, not 19. Make sure to use conductive grease (Penetrox) and lots of pressure at the feedpoint, to ensure that it can handle moderate current. The antenna can be stuck down the back of the shirt, carried, or easily mounted on bicycles, etc. With a little more weatherproofing, it will make a fine dual-band base station antenna of moderate gain (2.2 dbi) and stealthy appearance (paint it the same color as your roof then clamp it to a vent pipe). How Does it Work? Fabulously! The ½-wave dipole, even quite near the body (which you can use as a reflector if needed), works so much better than the H-T's duck - there's just no comparison. With the six-watt H-T on a full 12.7 volt, 7AH lead-acid battery, and the dipole, it performs as well as a mobile of the same power. I can walk around with this rig and hit several repeaters 20-30 miles away with ease with 4 watts - and even get full quieting into a repeater 9 miles away on ½ watt! August 2006 Update Since the QST article, many people have written emails to me regarding this design. Several are building variations of it, and we are compiling more data, which will be included in this article. Notes: Many people asked me for a formula for the offset. I don't have one. The antenna described herein has the feed point about 8.5% below center. This will vary with the coax and PVC used. How to determine the offset: Build one with the foil tape elements a few inches too short. Wrap aluminum foil around the ends, with rubber bands. Move the foil ends up and down till the SWR comes into line. Measure. Peel off the tapes and apply new ones, cut to the proper length. Test it a final time and if ok, seal the feed. That's how I did it. Don't use gray PVC. It just won't work - something about the plastic. —73, KV5R Page 3 © 2010 by KV5R — Rev. Dec. 1, 2010. Shop for soap making supplies here. Goals Acquire better soap-making skills. Make a small batch of 3-oil soap. Make exactly the right amount to fill a 6-cavity mold. Why? The first batch, 100% olive oil, didn't make good suds, and was too soft. Learned that olive needs about 20-30% coconut for cleansing and 20-30% palm for creamy suds. I want to make some soap good enough to give to friends. I want to try a silicone mold. Tools and Supplies Used Same as the first batch, with the addition of coconut and palm oil, and a silicone mold that makes six 3x3x1¼-inch square 6-ounce bars. Tools and Supples Used Recipe Classic 3-Oil Soap; recipe developed on soapcalc.net (by me) after reading several articles on 3-oil soaps. This time, I used grams instead of ounces to get better resolution from the little scale. My new mold has a total volume of 6x9x1¼ inches, so I then used the summerbeemeadow.com calculator, which allows one to re-size the batch to fit a certain rectangular mold volume (won't work with odd-shaped cavities; see below). 50%, 402g. Olive Oil (Walmart) 25%, 201g. Coconut Oil 76° (wholesalesuppliesplus.com) 25%, 201g. Palm Oil (wholesalesuppliesplus.com) 115g. NaOH × 2.333 = 268g. water = 30% caustic solution 5% Superfat (SF) Unscented. Procedure Now that I've written the articles on the previous 6 pages, I'll test my assumptions by following my own procedures (more or less). Weigh water into Pyrex cup. Chill water down to 60°F. Weigh lye and mix with the cold water (with cup in sink water and ventilation running). Weigh each oil into SS pitcher and put on stove to heat. Stabilize both temperatures at 115°F. Stick-blend until full trace. Pour into cavity mold. Crafters Choice™ Silicone Mold #0605 (wholesalesuppliesplus.com). Makes (6) ~6 oz squares. Cover and insulate overnight. Clean-up. Cure 4 weeks. Make fancy old-fashioned wrappers on the computer and print on fancy cotton-bond stationery paper. Total time: about 1 hour, including pictures and clean-up. Conclusion: The process went well; trace was again much faster than expected; and my volumetric calculations were almost correct: I had 1-2 fl. oz. too much for the cavity mold. Perhaps a better way to calculate soap volume: The specific gravities of vegetable oils is about .88 to .93, so to simplify let's call it .90. We can estimate the soap's final volume by taking the oil weight, 28.36 ÷ .9 = 31.51, + the water weight, 9.46 ~= 40.9 fl oz. My cavity mold will hold 38 fl oz, so I want to make 39 fl oz of soap (~1 oz left over clinging to pot and utensils). If we decide that 9 oz of water is close enough, we can subtract that, and need 30 fl oz of oils, and since we say the oils' S.G. is .9, 30 × .9 = 27 ounces (weight) of oils. I used about 28.5, and indeed had about 1½ oz too much, so maybe the .9 S.G. is a good figure. I'll do it that way next time and we'll see. Tomorrow: Getting bars out of the silicone mold was harder than expected. The problem is stretching the mold enough to get air under the soap without damaging it. Hot water on the bottom helped. Also, the soap was too thick when I poured, and it held some bubbles around the sides of the mold. So how does it work? Tried a bar after 12 days curing and it's very nice, much like Ivory, as expected. Good hardness, good suds, and good cleaning. Rinses easily with no oily film. No wonder that recipe is called classic 3-oil. Pictures 268 grams of water... Water cup in sink water... 115 grams of lye... Dump the lye into the water... (You really should use rubber gloves here.)Page 4 © 2012 by KV5R — Rev. Jan 7, 2012. Shop for soap making supplies here. Goals Design a soap with high-cleansing, creamy lather, and moderate conditioning properties. Make a video of the whole process, for those who learn best by watching. Soap Recipe Designer Spreadsheet See full-size PDF Spreadsheet The plan was to push the cleansing way up using by more coconut oil, while getting reasonable conditioning and creamy lather. Although the conditioning and lather look pretty low on the chart, I've found this is the best soap I have ever made. The Video Note that the video is edited for time—it actually took about 45 minutes to do, including clean-up. The video runs just under 15 minutes, including 2½ minutes of introduction. Results I let it cure on the rack for six weeks, then started using it. It's the best soap I've ever made! Strong on cleansing, but not harsh; great creamy lather; and good conditioning. Doesn't dry the skin or itch. Concluding Thoughts So there's my sixth batch, and my first soap video. What did I learn? It's not easy to make soap and a video at the same time! The commentary was unscripted, and I made a few boo-boos with camera angles, etc. Using nearly equal amounts of coconut, olive, and palm oils makes an excellent soap. It's a real pleasure to use! Stay tuned for more in the coming weeks. So until the next batch, good soaping! —KV5R Page 5 © 2011 by KV5R — Rev. Jan 1, 2011. Shop for soap making supplies here. Goals Design and make a well-balanced soap with one fat and 4 oils. Try a Fragrance Oil for the first time. Test my new soap recipe designer spreadsheet. Take more pictures (below) and write ya'll up another web page. My New Soap Recipe Designer Spreadsheet I spent the past week completely rebuilding the Chris Mathes Soapsheet in OpenOffice Calc. My new version includes fat/oil properties (like soapcalc.com), with a normalized bar graph. Updated all the fat/oil/butter prices from extensive web searching. The new sheet also has dynamic scaling and soap volume calculations. Once I had the 'sheet working right, it was lots of fun to change oils and weights, and watch the properties change on the graph. It instantly shows the interactions of the various oils' properties, making it fairly easy to find a balanced mix, or push around any desired soap property. KV5R's Soap Recipe Designer Template for OpenOffice Calc I started out designing a 4-oil recipe with what's on-hand, including fractionated coconut oil. It just would not balance its properties. Then I added the lard and things started looking better. After another hour of fine tuning I hit upon the nice balance of properties that you can see in the graph. Some soapers say that animal fat soaps clog your pores or are otherwise undesirable. Baloney. Once any fat/oil is saponified, it is no longer what it was—it's now soap, not fat. The lard is cheap, and it helped balance properties the oils were missing. Here's a shot of all the stuff I put in this batch: Ingredients The Recipe Developed using my new & improved soap recipe designer, with the goal of balancing all the properties near mid-range. 270 gm (34.6%) Armour Lard 115 gm (14.7%) Crisco soybean oil 140 gm (17.9%) Crafter's Choice Castor oil 105 gm (13.5%) Crafter's Choice Palm oil 135 gm (17.3%) Crafter's Choice Coconut oil, fractionated 14 gm (1.8%) Crafter's Choice "Baked Bread" fragrance oil 111 gm NaOH × 2.33 = 259 gm water = 30% caustic solution (a 3% water discount) 5% lye discount Target volume (at pour): ~1100 ml to fill the 1075 ml 6-cavity mold and allow a little waste. The Proposed Procedure This time I'll get the lard and oils heated, weighed, and mixed first, then put all that oil stuff away, then make the caustic solution. I have such limited counter space! Weigh the lard into a bowl and microwave it for a minute. Stir. Microwave 30 seconds more. Now it's clear. Warm the palm oil in the microwave for a minute. Shake bottle. Tare the pot and add the oils, weighing and taring each. Weigh the water in Pyrex measure and put in sink cooling & safety water. Weigh lye in a dry bowl and mix with the water. Stir. Avoid a few fumes. Warm the oils and cool the caustic solution to around 115°F. Pour the 30% caustic solution into the oils. Mix and stick-blend until light trace. Add the fragrance oil. Stick-blend to medium trace. Make sure fragrance is well-mixed. Pour into polyiso-insulated silicone cavity mold. Smooth top, cover and insulate. Clean up. Next day, un-mold and cure 4 weeks. Total time: about 1 hour, including pictures and clean-up. Results The process went very well. From past experience, I planned to pour sooner and make prettier bars with smooth tops and no side bubbles. It took about 2 minutes to light trace and, after the FO, about 20 seconds more to medium. It was at pudding thickness by the end of the pour. This time, my volumetric calculations were perfect! The last bar was level with a good pot scraping. Checked two hours later. Yep, it's doing a nice warm gel phase. The weight on top of the polyiso insulation has the top of the soap nice and smooth. The baker's parchment is a good mold cover: it lets out air bubbles, and doesn't stick to the soap (but it wrinkles later). These are gonna be perfect-looking square bars, requiring little or no trimming. Checked 6 hours later and it's still in gel! The polyiso box works much better than a big towel, and is much easier to handle. Pictures And so thare I was, justa waaaayin' out them thare hawg rendrins... Hard oils (lard, palm) heated up; all lined up in list order... Weighing the water... Weighing the lye... (oh my!) Dumping it in the water (carefully), then stirring it... Instant hot caustic—exothermed to 175°F. Avoid the fumes! Weighing 14 grams of fragrance oil... (Yes, I tared the syringe first.) Why 14? That's 1.8% of the oil weight—3% is nominal, but I want it very light.Page 6 © 2010 by KV5R — Rev. Nov. 24, 2010. Shop for soap making supplies here. Goals Acquire soap-making skills. Make a small batch of unscented pure castile (olive oil) soap. Why? I hate commercial detergent. It gets through my skin and I can taste it for hours. Yuk. Allergic to lauryl/laureth sulphates used in most "skin care" products. Itchy. Hypersensitive to most fragrances used in commercial products. Can't breathe in the soap aisle, much less the shower. Tools and Supplies Used No endorsements here—this is just what I got and where I got it. Digital kitchen/postal scale, Walmart, ~$20. Stick blender, Oster 250-watt variable-speed, Walmart, ~$25. Small dial thermometer, Walmart, ~$8. Walmart GV Pure Olive Oil, 25.5 fl.oz, ~$4 (not Extra Virgin). Roebic Crystal Drain Opener (100% Lye), 2 lb., local hardware store, ~$16. (See essentialdepot.com for lye and potash.) Totals: Tools: $53; supplies: $20; supples used: ~$6 for ~32 oz. (907g) of soap (8 bars ~4 oz. (113g) each). Already had: 3 qt. SS pitcher (like restaurants use for serving ice water). 2 cup Pyrex measuring cup (for mixing lye). 4x12 rectangular Tupperware with lid (#892-4) — my on-hand loaf mold. Recipe Don't use it—keep reading! 25.5 oz. bottle of olive oil (see "Oops!" below) 8 oz. lemon balm herb tea, strained and cooled (waste of time: didn't add any lemony smell to soap, but did make interesting marble pattern. Next time I'll just use distilled water). 3.30 3.02 oz. NaOH crystals, per MMS calc, 5% superfatting. Procedure Made the lemon balm herb tea, about 12 oz. (boiled in microwave then whizzed with stick blender). Strained and measured 8 oz. of the tea (on the scale) into Pyrex measure; set in freezer to chill. Run sink 1/3rd-full of cold water. Measure 3.3 oz of lye powder into very dry bowl on scale. Used too much lye. Set Pyrex of cold tea in sink water (making sure it was not about to float or tip over). Added lye to tea and mixed. Fumed for just a few seconds; not a problem. Temperature rose to 170°F (77°C). Should have started with colder water. Put opened bottle of OO in the microwave for 1 minute. Rose to 120°F (49°C), no fire needed. Poured warm OO into a 3-quart SS pitcher. Forgot to weigh it. Let both caustic solution and OO cool to 115°F (46°C). Poured caustic into oil and stirred a few seconds with SS spoon. Inserted the stick blender, at a slight angle, and fired it up on lowest speed. Nice rolling churn; no air whip. Light trace in about 1 minute. WOW! 30 seconds later it was thick pudding! Poured, scraped and patted into 4x12-inch Tupperware. Popped on the lid and wrapped it in a towel. Clean-up: hot water and dish detergent in the SS pitcher. Whiz it with the stick blender. Finish up with soapy sponge and then rinse well. Total time: about 30 minutes, including clean-up. Conclusion: It was a whole lot easier and faster than expected. This would have taken several hours without the stick blender (olive oil is notoriously slow to trace by hand). About 5 hours later, I unwrapped it for a peek. White at the corners, and a white film on top. Most of it is gel. Looks like I made a fine batch of axle grease! Oh well, we'll see tomorrow. Tomorrow: Solid but not crumbly; soap brick popped right out of the polyethylene Tupper. Cut easily with a cleaver, like a medium-firm cheese. So how does it work? Washed hands with shavings. Mild, very little lather, rinses easily. But nowhere near as nice as Dr. Bronner's liquid castile. I guess I'll make a new and better batch! What Went Wrong Oops! I forgot to weigh my oil! I just used 25.5 on the bottle, but that's fluid ounces. Looking up the specific gravity of olive oil, .92, that means my oil weighed 23.46 oz. Plugging that weight into the MMS calc, I should have used 3.0 oz. of lye, not 3.3. No wonder it full-traced in 1½ minutes! Oh well, I have some slightly caustic soap. Next day I decided to rebatch it and add 2-oz. more oil, but didn't know what I was doing, and messed it up. Now it's a spongy goop. Using herb tea for the caustic mix was a bad idea. Should have used distilled water and added powdered herb during trace. Should have used colder water. Took too long for the caustic to cool down from 170 to 115. Had to reheat the oil twice. Didn't know that 100% olive oil makes lousy soap. Too soft, and almost no lather. That's what coconut and palm are for! Photos Sorry I didn't get any during the process, but who wants to handle a camera and lye at the same time? These pictures were later staged. Ingredients Water in Pyrex in Water in Sink (saucer keeps cup stable) Tare the bowl... Weigh the lye... Now cap it! Collect any stray crystals with a damp sponge, then sink it. Dump the lye in the water... Stir... This is when it will heat up and fume for a few seconds. Notice the bowl, with a few clinging crystals, went straight into the sink water.Page 7 © 2010 by KV5R — Rev. Dec. 12, 2010. Shop for soap making supplies here. Goals Acquire better soap-making skills. Make a soap with pine tar—extremely fast to trace and set, requiring planning, speed, and very cold ingredients. Make a batch of soap suitable for camping or working in the woods. It needs to clean well, and leave a scent bugs can't stand. Make exactly the right amount to fill my polyiso-insulated silicone mold. The Stuff Same as the second batch, with the addition of pine tar, citronella, cornmeal, and an insulation box I made for the silicone mold. I used soybean oil instead of olive this time, because it has almost exactly the same soaping properties as olive oil, at one-fourth the cost. Ingredients The Recipe Developed using the Chris Mathes Soapsheet, after reading several articles on pine tar soaps, citronella soaps, and (you guessed it) cornmeal soaps. 300gm. (38%) Crisco "Pure Vegetable Oil" (soybean oil) 200gm. (26%) Coconut Oil 76° 200gm. (26%) Palm Oil 80gm. (10%) Bickmore Light Pine Tar (TSC) 25gm. Citronella oil 42gm. Hard Cornmeal (made by grinding and sifting popcorn kernels—and yes, it was way too scratchy!) 106gm. NaOH × 2.0 = 212gm. water = 33.33% caustic solution (6.33% water discount) 4% Superfat (SF) The Proposed Procedure Due to the additives, this time I'm gonna get all the oils and additives measured and mixed first. The caustic will be made with chilled water, then both it and the oils will be chilled down to 60°F before mixing. Pine tar makes soap react so fast that many soapers say use room-temp ingredients, use full water (lye × 2.7) and don't use a stick blender. I plan to use chilled ingredients and a 6.33% water discount—and the stick blender is needed to keep the cornmeal aloft (it settles very fast). Chill water and cup down to 40°F. in freezer. Heat the coconut and palm in the microwave (16 oz, 1 min.) so they'll pour easily. Weigh the pine tar into a disposable container and microwave it a bit (20 sec.). Weigh each oil (and pine tar, citronella, cornmeal) into SS pitcher, whiz, and put in freezer. Weigh lye and mix with the water, cool to 90 in the sink, then put in freezer. Stabilize both temperatures at 60°F. Pour the 33.3% caustic solution into oils. Mix and stick-blend with very short bursts until medium trace. Quickly pour into prepared, insulated silicone cavity mold. Cover and insulate. Clean-up. Next day, un-mold and cure 4 weeks. Total time: about 1 hour, including pictures and clean-up. Results The process went fairly well, but trace was again much faster than expected. Even with 60 degree ingredients, it got thick in about 1 minute—it nearly seized as I was molding it! This time, my volumetric calculations were almost perfect, but it thickened so fast that the waste was hard to recover, and one bar will be a tiny bit short. I should have started pouring about 15 seconds sooner. After a couple hours, I stuck the thermometer through the insulation to check it's gel-phase temp. Surprisingly, it was only about 95 degrees, and quite firm. So it's not gonna gel? Maybe it needs some help. I took the mold out of the insulation box and put it in the oven at about 140°F. According to what I've read, 4 hours at 140 will finish saponification—but OH! that smell! Not the pine tar, but the citronella! It's burning my eyes, and it's too cold outside today to keep the door and windows open. I'm thinking 25gm was too much. Anyway, after about 4 hours in the oven I took it out to the tool-shed and aired out the house. Whew! Tomorrow: Unmolded easily, and now that it's cold, the citronella scent is light and tolerable. Little or no pine tar smell. The soap is a dark brown, about like coffee. The cornmeal didn't settle during the cook. So how does it work? Took a shower with it a few days later. Creamy white lather, moderate citronella scent, and the cornmeal is slightly scratchy, but I should have ground it finer and used more. Rinses easily and doesn't leave any scent on you. May repel bugs during use, but not after. Overall, I'm a little disappointed. I'm thinking maybe this should be a hot-process soap, with the pine tar and EO added after cooking, so they're not part of the reaction. Pictures Here's my ¾-inch polyiso mold insulator. Corners are just taped; contact cement would be better. #1 12-16-2018, 02:22 PM (This post was last modified: 01-14-2025, 11:04 AM by iz2bkt.) Guide to using IC7100BKT with WSJT-X, JTDX, MSJV: IC7100BKT Version 1.13.0 and later: - In IC7100BKT go to Utilities -> Configure - Enable LAN Protocol on port 52002 - Confirm with OK In WSJT-X or JTDX: - Go to File -> Settings -> Radio - In Rig select DX Lab Suite Commander - Confirm with OK In MSHV: - Go to Options -> Interface Control - In the Network box -> Server type 127.0.0.1 and in Port type 52002 - In RIG select DX Lab Suite Commander IC7100BKT versions prior to 1.13.0: - Download and install BktNetInterface - Start BktNetInterface - Select the CAT in use (IC7100BKT) - If you want the connection to start automatically when you open the program, check "Auto-start in system tray" - Click "Start" - Start JTDX or WSJT-X - Go to "File" -> "Settings" -> "Radio" - In "Rig" select "DX Lab Suite Commander" - Close with "OK" To start BktNetInterface with the CAT: - Start IC7100BKT - Go to "Utility" -> "Startup" - In "Description" write "BktNetInterface" - In "Program" select "BktNetInterface.exe" from the installation folder - Click "Save" It is recommended to start IC7100BKT first and then JTDX, WSJT-X or MSHV. 73 de Mauro, IZ2BKT WSJT-X implements communication protocols or "modes" called FST4, FST4W, FT4, FT8, JT4, JT9, JT65, Q65, MSK144, and WSPR, as well as one called Echo for detecting and measuring your own radio signals reflected from the Moon. These modes were designed for making reliable, confirmed QSOs under extreme weak-signal conditions. JT4, JT9, and JT65 use nearly identical message structure and source encoding (the efficient compression of standard messages used for minimal QSOs). They use timed 60-second T/R sequences synchronized with UTC. JT4 and JT65 were designed for EME ("moonbounce") on the VHF/UHF/microwave bands. JT9 is optimized for the MF and HF bands. It is about 2 dB more sensitive than JT65 while using less than 10% of the bandwidth. Q65 offers submodes with a wide range of T/R sequence lengths and tone spacings; it is highly recommended for EME, ionospheric scatter, and other weak signal work on VHF, UHF, and microwave bands. FT4 and FT8 are operationally similar but use T/R cycles only 7.5 and 15 s long, respectively. MSK144 is designed for Meteor Scatter on the VHF bands. These modes offer enhanced message formats with support for nonstandard callsigns and some popular contests. FST4 and FST4W are designed particularly for the LF and MF bands. On these bands their fundamental sensitivities are better than other WSJT-X modes with the same sequence lengths, approaching the theoretical limits for their rates of information throughput. FST4 is optimized for two-way QSOs, while FST4W is for quasi-beacon transmissions of WSPR-style messages. FST4 and FST4W do not require the strict, independent time synchronization and phase locking of modes like EbNaut. WSPR mode implements a protocol designed for probing potential propagation paths with low-power transmissions. WSPR is fully implemented within WSJT-X, including programmable "band-hopping". Latest General Availability (GA) releases: WSJT-X 2.7 WSJT-X 2.7 introduces a new program called QMAP, new Special Operating Activities Q65 Pileup and SuperFox mode, an option to Update Hamlib at the click of a button, a new program feature Message System, and a number of other enhancements and bug fixes. Changes from earlier versions, and in particular from version 2.6.1, are described in the Release Notes. If you will use the Q65 mode, please read the Quick-Start Guide to Q65. On Windows platforms, WSJT-X 2.7 also includes MAP65 3.0, a wideband polarization-matching tool intended for EME. If you will use MAP65, be sure to read the Quick-Start Guide to WSJT-X 2.5.0 and MAP65 3.0 SuperFox mode behaves operationally like the old-style Fox and Hounds mode but uses a new constant envelope waveform for Fox's transmissions. Messages can be transmitted simultaneously to as many as 9 Hounds with no signal-strength penalty, resulting in a system gain of about +10 dB compared to the older Fox-and-Hound operation with 5 slots. Further details on SuperFox mode can be found in the SuperFox User Guide. Upgrading from a previous version will be straightforward. There is no need to uninstall or move any files. If you want to make sure to have the latest list of default working frequencies, go to File | Settings | Frequencies, right-click in the Working Frequencies list, and select Reset. Documentation: The WSJT-X 2.7.0 User Guide is available online. This document should always be your first source for help. Use your browser's search facility to find a keyword or topic. German translations of two associated documents by Enrico Schürrer, OE1EQW: If you plan to use FT8 DXpedition Mode, be sure to read the FT8 DXpedition Mode User Guide. Versions of WSJT-X labeled with a "-rcx" suffix, for example WSJT-X v2.2.0-rc6, are Release Candidates sometimes offered temporarily for beta testing purposes. You should upgrade to the GA release when it becomes available. The -rc# program versions are not suitable for long-term general use. Installation packages for WSJT-X 2.7 Windows: Linux: Installation instructions for Linux can be found here in the User Guide. Download the package file appropriate for your system, from the list below. (Versions installable with "apt-get" and "yum" will be made available as soon as our package maintainers create the packages.) Note: these packages are unlikely to install properly on Linux distributions with required dependencies at lower versions than those on the named distributions. In such cases building from source is the correct way to install WSJT-X. Macintosh macOS: Installation instructions for version 2.7.0 can be found here in the User Guide. Version 2.7.0 for macOS 10.13 through 15: wsjtx-2.7.0-Darwin.dmg Source Code: WSJT-X is licensed under the terms of Version 3 of the GNU General Public License (GPL). Development of this software is a cooperative project to which many amateur radio operators have contributed. If you use our code, please have the courtesy to let us know about it. If you find bugs or make improvements to the code, please report them to us in a timely fashion. Build and installation instructions are in the INSTALL file inside the tarball. Source code and necessary resources for WSJT-X 2.7.0: wsjtx-2.7.0.tgz ©2001-2025 by Joe Taylor, K1JT I know it is only three days since the last posting with almost the same title, but I seem to feel the need to rant on, and introduce MSHV. I am a big fan of Joe Taylor's WSJT modes. I use the software he publishes a lot - for WSPR, JT65 and JT9 in particular. These modes are included in the latest WSJT-X version (I use v1.6.1). WSJT-X is the wrapper used to include all the current modes and it can be used with CAT control to set up your rig frequency and so forth. We are promised that gradually all the new "JT" modes will be included in WSJT-X offering a seamless operating experience. This is great but operators on the bands are not keen to switch to new modes. ISCAT for 6m and JTMSK for higher VHF bands, are both on offer in v1.6.1, but almost nobody uses them. Instead the bands are full of JT6M and FSK441, modes which are now rendered obsolete by ISCAT and JTMSK. As a result, most operators on the VHF bands use older versions of WSJT - WSJT10 in particular, to generate their signals, and those older pieces of software do not support CAT. The consequence of all this is that using WSJT10 via a USB connection with the Icom IC-7100 and 7300 is almost impossible without some additional software such as CAT7200 to intervene and convert things. I am pretty loyal to WSJT. However, it was made as open source software for a reason, to allow others to package those modes as they thought fit, and thus stimulate development. Frustrated by the lack of software which can produce FSK and JT6M and work with the Icom rigs using digital audio streams, I have been checking out MSHV. Greg, SP3RNZ, suggested this software on his blog - see comment in my last posting here I had to test MSHV before recommending it myself. I did try it last year in an older Beta version (v0.99) when Chris GM4ZJI suggested it, but now it is in a much more advanced state in v1.13. There are 32 and 64 bit versions and I have tried both. It works fine with both the IC-7100 and the IC-7300 without the need for USB SEND, CAT7200 or anything else. MSHV has proved to be a complete success here so far. Here is a screen shot of me doing some listening on 144.370MHz I put out a CQ using MSHV earlier and worked DL4DWA in JO61 (1138km). A good little test, and a quick 6 minutes QSO was the result. MSHV seems to put very little load on the processor and so far has not crashed or lost contact with the rig. It is also very nice to be able to include the working frequency without having to insert it manually every time -MSHV provides a handy box to include this feature. LZ2HV, creator of MSHV, says that he took the open source software from WSJT and re-wrote it in C++. Presumably it includes the latest hamlib settings as both the Icom rigs are in the list of supported equipment. MSHV includes JTMS and ISCAT as well as FSK441 and JT6M. You can download it from here . Installation is easy, but do not make the mistake I made - once you have downloaded MSHV and removed the files from the ZIP encoding, the .exe applications with their pretty icons look as if they can be clicked and moved onto your computer's desktop. No! You need to create a shortcut and move that to the desktop. If you make the silly mistake I made, the software will work OK but you will have to re-enter the audio, interface and macro settings every time you open it. On the other hand if you create a shortcut these will be remembered each time. Stupid mistake to make. You would think I would have learned by now (I made the same mistake last year with the earlier version). I can see the issue for the WSJT team. To spend valuable development time incorporating "outdated" modes into WSJT-X may seem like a waste of resources. But, maybe to spite them, most amateurs are still using FSK441 and JT6M. This arose before - JT6M was dropped from WSJT9 and replaced with ISCAT. By version 10 JT6M had been re-introduced. The best way for this sort of problem to be resolved would be for amateurs to download the latest version of WSJT-X and use the modes provided. But amateurs are resistant to change. They like using outdated modes. Look how many use RTTY, a spectrum consuming, insensitive, slow, power wasting dinosaur from the days of mechanical printers. And as for AM ... but amateurs don't change easily. I thought this is supposed to be a scientific hobby, not a museum for preserving ancient technology. For as long as people are using FSK441 and JT6M and refusing to use JTMS and ISCAT this problem will continue. Even then I saw a report that Joe Taylor had suggested that JT9 would be the preferred mode to develop further for MS use. I can see why, as the advanced versions of JT9 are terrific, but this is a step further beyond even ISCAT and JTMS. If most amateurs cannot even make the first step, when will they make the second? There is no point me using new high speed versions of JT9, as I have tried to do, only to find everybody else is using old style JT6M. Luckily, for as long as we stick to FSK and JT6M, good people like LZ2HV will be willing to invest their time and effort into making software that allows us to continue. I for one appreciate his efforts in producing a very good re-formulation of WSJT modes. I wish it was different. I wish we would set a date and change over to the latest modes produced by the latest efficient software. We could also change our working frequencies to fit into the bandplan - for example use 50.320-50.380 and 70.250 as the bandplan suggests. In the real world I know this is not going to happen. I think that the present situation just devalues the bandplan. At the moment the bandplan sets out clearly the frequencies and we all ignore them. Until that great conversion day dawns, when we all upgrade to the best and fastest JT modes available, I suspect MSHV will have a large band of fans. Including me I guess. Please cancel the suggestion, made by me just 3 days ago, that I was sticking with WJST and CAT7200. I am modernising and adopting MSHV. 73 Jim GM4FVM We had a great presentation today by Wendell, N4VLK, on the digital mode FT-8 today at the CARS club meeting. I was asked if I have run FT-8 / WSJT-X on my IC-7100. Yep, I have, and it works quite well too. I know where are a number of guys in the club that have the IC-7100 so I thought I would share my setup information. Here is a very quick guide on setting up WSJT-X on the Icom IC-7100. I am assuming that you have already installed the Icom supplied USB CAT and USB Audio drivers. If not, do install them before continuing with the WSJT-X installation. First, download and install Dimension 4, an app to keep your clock synced to the NBS time source. This is very important to make sure your receive and transmit timing is correct, otherwise WSJT-X will not decode signals and other stations will not be able to decode your signal. (you can click on the images below to see a larger view) Website for Dimention 4 Installation is pretty straight forward, select a sever to sync with and click the Sync "Now button." Dim 4 Install screen Once you have Dimension 4 installed, go to the website for WSJT-X and install the Windows version. Download site for WSJT-X You can use the defaults for the install. Run the WSJT-X program and you should see two windows, the waterfall display window and the main window. In the main window click "File" on the top menu and select Settings from the pull down. On the General Settings tab, enter your Call and your Grid Square information WSJT-X General Settings Then on the Radio tab enter the info for you CAT port for the IC-7100. Select the radio type, then the serial port, in my case it is Com 6, and select CAT for the PTT. Once these settings are complete you can click on the "Test CAT" button at the bottom right of the window. If button turns Green your CAT is setup correctly. You can also test the PTT function with the "Test PPT" button. WSJT-X Radio Settings Tab Then go to the Audio Tab and select the USB Microphone and Speakers for the audio interface. WSJT-X Radio Tab Settings A couple more settings and we are done! Now go to the Reporting Tab and set the program to send data to PSKReporter website. WSJT-X Reporting Setup Close the Setup window and one final setting is to the "Mode" on the main menu bar of the WSJT-X main screen and set the mode to FT-8. Can also run other modes with this program if you desire. WSJT-X Mode Setting Ok, you should be ready to operate at this point! I suggest watching for a while to get a feel for what is going on with the program. Also, reading the user guide will be a great help. Have fun on FT-8! If you have any questions that I can help you with email me at bob@n4rfc.com. Update – 5/13/2020 – I have had several requests to add information on setting up the menus in the 7100 for digital modes. Below are screen shots of the "Connectors" screens on the 7100. To get to these settings, press the "SET" button on the front panel, then scroll to the "Connectors" and touch it then you can scroll through the entries. Everything you need to get the digital mode transmitting is in that setup screen. (The photos below are thumbnails, click on each one to see the full size photo) 73 and GL