CTR2-Dial

Firmware Manual

v2.04.01

Last Revision: February 12, 2026

Updated to firmware v2.04.01

Revised sections for this version are highlighted in yellow

Contents

Introduction

CTR2-Dial firmware is a custom MIDI solution that runs on the M5Dial from M5Stack. It was designed by a ham, for ham operators. It is based on the popular CTR2-MIDI controller's firmware, also available from Lynovation.com. It supports both USB and Bluetooth-LE MIDI connections and is used with radio control programs like Marcus Roskosch's (DL8MRE) apps for iOS and MacOS. It's also at home working with Thetis, Simon Brown's SDR-Console, and SparkSDR to control your Apache Labs ANAN or Hermes Lite 2 (among others) on your Windows computer. It's also a great addition for those that use Remote Ham Radio and RemoteTx. It can even be used as a general-purpose MIDI controller.

<<< IMPORTANT >>>
The default MIDI control labels in **CTR2-Dial v2.04.00** have been modified and *do not match the default MIDI map shipped in SmartSDR for iOS/MacOS*. You must import the SmartSDR map from the CTR2 Configuration Files page on Lynovation.com, or manually edit the map in SmartSDR to match the new labels in **CTR2-Dial**. More information on editing the control labels can be found here. Information on editing the map in SmartSDR can be found here. I also have a video on YouTube explaining the mapping process.

CTR2-Dial Firmware is Now Part of CTR2-Flex Firmware

The **CTR2-Dial** firmware runs on the latest **CTR2-Uno**, **CTR2-Duo**, and **CTR2-Quad** hardware. It also runs on the older **CTR2-Dial** and **CTR2-Flex** hardware. **CTR2-Dial's** companion firmware, **CTR2-Flex**, used to directly control a Flex 6000/8000 radio, also runs on these hardware platforms.

> **NOTE:** To minimize the confusion caused by the previous naming convention, **CTR2-Dial** hardware is being depreciated and **CTR2-Flex** hardware was rebranded at the start of 2026 to **CTR2-Uno** and **CTR2-Duo**.

Since both firmwares run on all M5Dial based **CTR2** hardware platforms, it made sense to integrate the firmwares into a single binary file that can dual-boot into either firmware. If you use one of these units to control a Flex radio this gives you two options:

- Controlling your radio through a 3rd party control app using **CTR2-Dial's** MIDI control firmware
- Controlling your radio directly (with or without SmartSDR) using **CTR2-Flex's** WiFi control firmware. Simply select the mode you want to run in the Settings menu. Your unit will run the selected firmware until you change it.

Both programs look and work the same as they did when they were separate. This just makes it easier for me to maintain, and more convenient for Flex users that have a need to run both firmwares without having to re-flash their unit.

If you don't run a Flex radio, just order your controller set to boot into **CTR2-Dial** firmware and use this manual for your reference. When you run the **CTR2-Flex** firmware, refer to the CTR2-Flex Firmware Manual for information on that firmware.

NOTE: With a few exceptions for added features, this manual can be used for the older single-boot **CTR2-Dial** v1.01.01 firmware too.

How to use this manual

This manual should be used as a reference manual. An expanded Help system if you will. Items in the Table of Contents link to their write up in the manual. I've tried to group things logically and have added hyperlinks so you can quickly jump to other sections.

Some have commented that I use too many hyperlinks in the text. I do this so you can go to any section in the manual and easily navigate back to the descriptions of each control without having to read the entire manual first.

As this document evolves, sections that have changed since the last update will be highlighted in yellow.

The version number of this manual will follow the latest released version number of the **CTR2-Dial** firmware.

Feel free to contact me if you have question about a certain feature or have ideas for future improvements. I love to get feedback on my work. My email address is good on QRZ.com.

Legal Notice

What would a manual be without a legal notice? Here goes…

- This is a hobby endeavor. Nothing is guaranteed! Use this device at your own risk!
- I will do my best to make sure you receive functioning hardware if you buy the assembled unit and will work with you if there is a problem with your unit on arrival.
- I cannot guarantee or warranty the hardware supplied in kits.
- I make no warranty that the firmware provided will perform up to your expectations or be suitable for your application. A lot of compromises had to be made to fit the small display so review this manual to be sure you're comfortable with the user interface before purchasing.
- Software bugs are a fact of life and I try to find and correct all bug reports to the best of my ability ASAP.
- **CTR2-Dial** requires a license key to be fully functional. License keys can be purchased separately if you build your own unit. They are included with any purchase of any M5Dial based **CTR2** controller from Lynovation.com.
- License keys are tied to your call sign. This allows you to use the same key on as many devices as you own. You are not allowed to include your license key on units you build to sell or give to others. They will need to purchase their own key.

Change Log

NOTE: You must reset your unit to factory settings (page 7 in the Settings menu) when updating v1.00.xx firmware to v2.xx.xx firmware in order to get the correct mapping. You will also need to update the mapping on your control app and export a new backup file. If you use EspressIF Flash Download tool, use the "Erase" option to reset the unit to Factory settings.

v2.04.01: February 12, 2026

- Updated Appendix B: Installing or Updating CTR2-Dial Firmware to include ESPConnect as the primary method to flash firmware to the M5Dial on PCs and Macs (sorry, not Linux).
- Added HID support to allow **CTR2** units running **CTR2-Dial** firmware to control RemoteTx stations. See the RemoteTx section for details.
- Fixed several bugs in the **Knob** menu that were causing the wrong Dial function to be displayed.
- The text editor page now works correctly when pressing and turning knob A to move the cursor position.
- Removed the second page of Dial functions from the **Uno's Knob** menu. There are now only 6 functions available in this menu. On **Duo** units there is a menu for each knob.

v2.04.00: January 30, 2026

- Revised this manual to focus on **CTR2-Dial** firmware. Hardware references have been moved to the **CTR2-Uno**, **CTR2-Duo**, and **CTR2-Quad** hardware manuals. Click these links to open these manuals.
- A new **configuration repository** has been added to the Lynovation web site under Download -> Configuration Files. **CTR2-Dial** and **CTR2-MIDI** configuration files and MIDI maps for various 3rd-party radio control apps can be found here.
- Added Ring: Fine-Tune mode to the *Settings* menu
- **CTR2-Flex** firmware is covered under a separate manual. This manual can be found here.
    - The following new features have been added to this version:
    - Support for **CTR2-Quad's** four knobs, two pushbuttons, and separate input jacks for paddles and key/PTT
    - Added the option to export just your unit's basic settings or its basic settings and all of your favorite frequency lists. Export just the basic settings when sharing your unit's configuration with others. You can also your files to the CTR2 group.

Changes to previous versions can be found in Appendix H

System Overview

CTR2-Dial doesn't "talk" to your radio directly. It uses MIDI commands to tell a radio control app what to do. The MIDI protocol was developed to control musical instruments. It's a very small, simple, and fast protocol. Many radio control apps are adopting MIDI to allow other programs to control their functions instead of proprietary CAT protocols.

The main advantage of using a dedicated MIDI controller like the a **CTR2** controller over using a mouse is that the radio control app doesn't have to be "in focus" on your computer for the **CTR2** unit to control it. Commands from **CTR2-Dial** go directly to the app that it is registered to. You can change frequency or other parameters using **CTR2-Dial** while you are entering log data or working in another program.

Another advantage of **CTR2** controllers over generic MIDI controllers is that **CTR2** controllers are custom designed and build for ham radio operation. They have built in paddle, key, and PTT interface that make it easy to support these functions if the radio control app you're using supports them. It also supports Bluetooth LE and USB connections so it is usable on a wide range of devices.

Like a mouse, **CTR2-Dial** only sends commands. It does not receive information back from the app or the radio. It has no idea what is going on in the "real world". The radio control app you're using manages the user interface and the control communications with the radio.

How CTR2-Dial Works with Your Control App

Before getting into the details of **CTR2-Dial's** firmware, it's essential that you have a basic understanding of what **CTR2-Dial** does, and how it works with your radio control app.

CTR2 controllers running **CTR2-Dial** firmware are MIDI controllers with a touchscreen display. The gray ring on the display is used to navigate the menus or can be assigned to control a *dial control* function. The firmware supports 18 virtual dial controls and 42 virtual buttons. Each of these controls are "hardwired" to a MIDI command #. The MIDI command # for each control never changes.

You can edit the label displayed for each control and change the type of MIDI command used for wheel and slider commands. The label is just a convenient way to let you know what a control is *supposed* to be changing in your radio control app.

In order for **CTR2-Dial** to control your app, you must "map" each MIDI command # from **CTR2-Dial** to a function in the app. Your app will have a method to edit its MIDI controlled functions.

> For example, MIDI wheel control #100 is sent by the first *dial control* in **CTR2-Dial**. This control is labeled "VFO" on the **CTR2** controller's screen when it's selected. In order for MIDI control #100 to change the frequency of your radio, it must be "mapped" to the VFO frequency control object in the radio control app.

Controls in the **CTR2** controller can be executed by touching the display, or they can be assigned to the physical knob(s) and switches on the device.

When you execute a control in CTR2-Dial, a 3-byte MIDI command is sent to your app. The MIDI command includes the MIDI control #, and depending on the control type, i.e. a direction for a wheel control, a value for a slider control, or an On/Off value for a button control.

Every app has a function map that you must configure separately. This map tells the app what function to execute when it receives a MIDI command. This map is different in every app. For Marcus' iOS/MacOS

apps you can find it in **Tools -> CTR2 Dial Controller -> Edit Mapping** page, in Thetis, navigate to **SetUp -> Serial/Network/Midi CAT -> MIDI** then click the *Configure MIDI* button.

IMPORTANT: The default labels for the controls for Map 1 in **CTR2-Dial** are setup for Marcus' SmartSDR for iOS/MacOS apps, but they do not match the default settings he ships in SmartSDR. You'll need to edit the mapping in SmartSDR to match **CTR2-Dial's** default settings. Map 2 is set up for use with **Thetis** but you'll need to map Thetis to match the labels in **CTR2-Dial**.

You can download the configuration and map files for various radio control apps from this web page.

Your App Executes the Wrong Control

If you execute a control on **CTR2-Dial** and the app executes the wrong function you must either edit the label in **CTR2-Dial** to match the function mapped to that MIDI control # in the app, or edit the app's map to execute the function shown by the label in **CTR2-Dial**. It doesn't matter which data you change, but they must match.

The 18 dial controls are assigned MIDI control #100 through #117. The MIDI control # is shown on the display just above the label. The Dial Editor allows you to edit the dial's label and MIDI control type.

The 42 button controls are assigned to MIDI control #1 through #42. Buttons are displayed in the Buttons menu. MIDI button numbers are not displayed due to size constraints. You can edit the label for each button in the Button Editor.

The table below lists the MIDI button # starting on Button menu page 1, in the order they're displayed.

Page	Buttons		Page	Buttons		Page	Buttons		Page	Buttons	
1	1	4	**2**	7	10	**3**	13	16	**4**	19	22
	2	5		8	11		14	17		20	23
	3	6		9	12		15	18		21	24

Page	Buttons		Page	Buttons		Page	Buttons	
5	25	28	**6**	31	34	**7**	37	40
	26	29		32	35		38	41
	27	30		33	36		39	42

I have posted a video on YouTube describing the mapping process. You can find it here.

NOTE: You can export **CTR2-Dial's** settings to your computer to back them up, and import them back in as needed. To do this, you must use a terminal program that supports XModem file transfer. Tera Term is the easiest terminal program to use and it supports XModem. Putty and Apple/Linux Terminal programs do not support XModem. More information on importing and exporting settings in **CTR2-Dial** can be found in the Backing Up and Restoring Your Settings section.

Multiple Controllers / Two Firmwares

As mentioned previously, two versions of firmware are available that run on all M5Dial based **CTR2** controllers.

CTR2-Dial firmware uses MIDI commands over Bluetooth or USB to control a 3rd party app such as SmartSDR for iOS/MacOS, SDR-Control, FT-Control, TS-Control, K4-Control, Thetis, SDR-Console, and many other apps that support MIDI control of their functions.

CTR2-Flex firmware on the other hand, is great for controlling your Flex 6000/8000 series radio locally running any version of SmartSDR (or even without the SmartSDR user interface). However, since it doesn't support SmartLink, it presents challenges for remote operation.

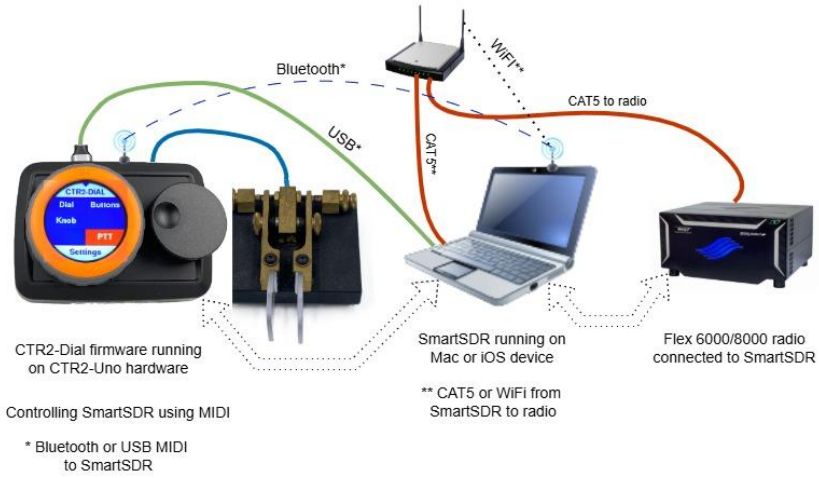
CTR2-Dial and **CTR2-Flex** firmware can run on any **CTR2** hardware based on the M5Dial. Starting with *Flex v2.00.00* firmware you can switch between the two firmwares without re-flashing your unit.

Switching Firmwares

To switch firmwares, open the **Settings** menu and navigate to the last page (page 9 on **CTR2-Dial** and page 11 on **CTR2-Flex**) and select *Run CTR2-Flex* or *Run CTR2-Dial*, depending on which firmware is running.

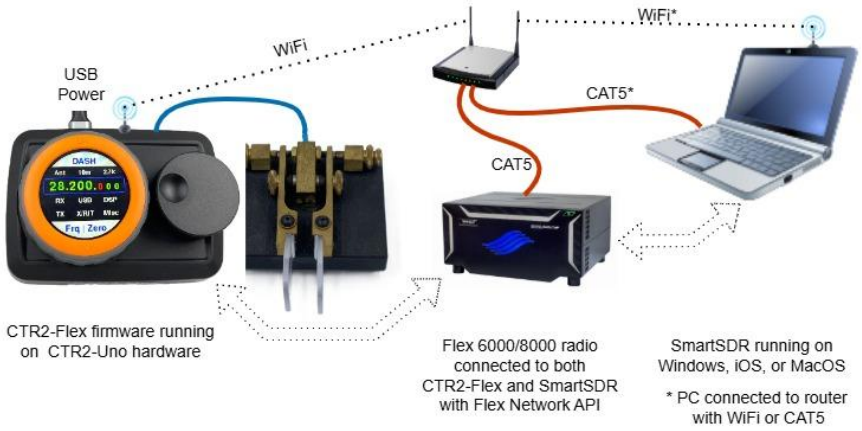
When using **CTR2-Dial** firmware, the controller connects to, and controls the 3rd party app that is controlling the radio.

This example shows a **CTR2-Uno** controlling SmartSDR for MacOS, which, in turn, is controlling a Flex radio. Any other 3rd party app that supports MIDI can be used.



For Flex radio users, **CTR2-Flex** firmware is great alternative for controlling your Flex radio locally running any version of SmartSDR (or even without the SmartSDR user interface). However, since it doesn't support SmartLink, it presents challenges for remote operation.

When using **CTR2-Flex** firmware as shown here the controller connects to, and directly controls the radio through the IP network using WiFi and the Flex Network API.



CTR2-Flex firmware includes a dashboard mode that presents radio information and allows control with or without SmartSDR running.

There are a few of trade-offs when running **CTR2-Dial** firmware on older **CTR2-Dial** hardware:

- **CTR2-Dial** hardware doesn't have encoder switches so you can't assign **Knob** switches to Button functions.
- **CTR2-Dial** dual-encoder hardware doesn't support the CW paddle input jack while all other M5Dial based **CTR2** controllers do.
- An upgrade kit is available for those wishing to upgrade their older **CTR2-Dial** or **CTR2-Flex** hardware to a **CTR2-Quad**.

Which firmware you use will depend on your use case. This manual covers the **CTR2-Dial** firmware. You can download the **CTR2-Flex** firmware manual here.

CTR2-Dial Features

CTR2-Dial runs on the M5Dial from M5Stack. The M5Dial features an ESP32-S3 processor, a 1.28" color touchscreen, and a built-in selector ring and hardware pushbutton. The firmware includes the following features:

- Supports USB MIDI and Bluetooth-LE MIDI connections
- 42 virtual button controls for modes/bands/filters selection
- 18 virtual dial (wheel/slider) controls for tuning, volume, RIT, XIT, etc.
- Assign the gray Ring encoder on the M5Dial display to any of the available **Dial controls**
- Up to four optional physical encoders (referred to as **Knobs**)
    - **CTR2-**Uno single-knob models have 6 wheel/slicer controls in the **Knob** menu
    - **CTR2-Duo** dual-knob models each have 6 wheel/slider controls available
    - On the **CTR2-Quad** unit:
        i. Knobs can be programmed to two functions (turn and push-and-turn)
        ii. Unassigned knobs control the selected *Dial* control and the selected control in the dashboard
- Two pushbuttons are provided on the **CTR2-Quad** – each of these can be programmed to any two button functions (using short and long-press) or be dedicated to control PTT.
- The gray ring on the M5Dial can be configured as a fine-tuning control (to make fine adjustments to the selected *Dial* control), or it can be assigned to always control the selected *Dial* function, or it can be assigned to always control any of the *Dial* functions in any page.
- Tuning can be locked to prevent inadvertent tuning
- 3.5mm (1/8") stereo input jack allows you to use your paddles to control the keyer, straight key, or PTT in the app (not all apps support all modes)
    - Separate jacks are provided on **CTR2-Quad** for paddles and key/PTT
- A virtual PTT switch
- Four beep volume levels, Off, Low, Medium, and High
- Automatic speed sensitive tuning & control adjustments
- Two maps allow **CTR2-Dial** to be used with separate programs without requiring configuration changes
- Seven color themes and backlight control
- Sleep and power down timers to save battery power
- Manual power down
- User programmable labels for each button and dial control
- Many power options – see the hardware manuals for each unit
- Turn the Bluetooth radio off to save 50 mA of current draw for battery operation
- Backup **CTR2-Dial's** settings using the **Import** and **Export Settings** option
- Firmware updating is done through the USB-C port
- Run **CTR2-Dial** or **CTR2-Flex** firmware without re-flashing the unit

Hardware

Hardware is covered in separate manuals for each model. Click the following links to open the hardware manual page for each unit: **CTR2-Uno** **CTR2-Duo** **CTR2-Quad**

> **NOTE:** To reduce confusion, older **CTR2-Dial** is being depreciated and the original **CTR2-Flex** hardware has been rebranded to **CTR2-Uno** and **CTR2-Duo**.

Registering the CTR2-Dial Firmware

You must enter a registration key for your **CTR2-Dial** if you don't purchase an assembled **CTR2-Dial** (i.e. you purchased the M5Dial separately). Registration keys are provided for free when you purchase an assembled unit from me. Users who prefer to source their own parts can purchase a registration key for the firmware at a minimal cost.

The firmware will run in **Demo** mode until registered. In this mode, the unit will automatically power down after 15 minutes. You can edit the settings and labels on the unit but you can't save or export them.

To enter your registration key, connect your unit to a terminal program such as Tera Term or Putty (see Appendix C, D, or E). Once connected, press any key to start the terminal server on the unit. Next, press the **DEL** key on the keyboard. Enter the call sign you registered and the registration key you received from me. The registration key is 8 hexadecimal digits (0 to 9 and A to F). Once registered, your call sign will appear on the splash screen at boot up, the program won't shut down after 15 minutes, and you'll be able to save changes in the maps and import or export backup files.

You can also use the on-screen editor to enter your registration credentials. Just select **Enter Call & Key** on the last *Settings* page. Turn the knob to change the letter and touch the screen (or press and turn the knob) to advance to the next letter position. Press the **OK** button to accept. The input starts with your call and advances to your registration key once the call has been entered.

I recommend writing your call and registration key on the unit's bottom label with a permanent pen. Cover it with a piece of transparent tape so it doesn't rub off. This way you'll have it if you reset your unit back to factory settings.

A dumb terminal can be used to edit the control's MIDI types and their labels, or you can use the **Edit Dial Setting** and **Edit Btn Label** options in the *Settings* menu.

MIDI Control Types

Before discussing the **CTR2-Dial's** functions, it's important to understand the types of MIDI controls **CTR2-Dial** supports. The table below summarizes each function.

Buttons	Application
Button controls	Buttons are implemented using MIDI Note controls. Each button in **CTR2-Dial** is assigned to a MIDI Note # starting at 1 and continuing to 42. When you execute a button function the program sends a MIDI Note ON command followed by a MIDI Note OFF command to the app for the assigned button.
Wheels & Sliders	**Application**
WheelA	This control is used with Marcus' iOS/MacOS programs, RHR, and RemoteTx for frequency control. This control is continuously adjustable and is centered at 64 counts. Frequency up/down control is initiated by sending a value above or below 64 depending on the direction the dial is turned. The offset from 64 counts determines the turning speed.
WheelB	This control is similar to WheelA and is used with Thetis and most Windows control programs. This wheel sends 126 counts to decrement the frequency by one step and 1 to increment the frequency by one step. When the *Dial* or *Knob* is quickly turned the firmware sends 10 commands for each step instead of one to speed up tuning.
WheelB-r	This control is the same as Wheel B except that 1 is sent to decrement frequency and 126 is sent to increment frequency. This control type was added to support the PI HPSDR control program.
SliderA	This control is similar to potentiometer. Its range is 0 to 127 counts. Turning the *Dial* control CCW reduces the count until 0 is reached. Turning it CW increases the count until 127 is reached.
SliderB	This control is similar to SliderA except that it is used specifically for center tuned controls like RIT and XIT. This control is centered at 64 counts. Decrementing below 64 counts applies a negative offset and incrementing above 64 counts sends a positive offset. When the value of this control reaches 64 when turning from either direction, **CTR2-Dial** beeps and pauses for 750 milliseconds. This allows you to find zero on the control. **HINT:** Touching the control's label in the center of the *Dial* display when the control is "off zero" resets the control to its zero position (64 counts).
Buttons Wheel Type	This control assigns a pair of MIDI buttons controls to send up and down button controls to the control program using the *Dial* or *Knob*. The lower control # is sent when the control is turned CCW and the upper control # is set when the control is turned CW. This is normally used for frequency or frequency tuning step up/down control. The control pair is determined by the Dial # as shown in the next table.

MIDI Button Pairs for Dial Button Mode

When the **Button** control type is selected for a **Dial** control, the MIDI button pair is determined by the Dial control #. Turning the knob CCW sends the lower MIDI command #, and turning it CW sends the upper MIDI command # from the table below.

Dial Ctrl #	Button Pair	Dial Ctrl #	Button Pair	Dial Ctrl #	Button Pair
100	50/51	106	62/63	112	74/75
101	52/53	107	64/65	113	76/77
102	54/55	108	66/67	114	78/79
103	56/57	109	68/69	115	80/81
104	58/59	110	70/71	116	82/83
105	60/61	111	72/73	117	84/85

EXAMPLE

To assign a knob control to increment or decrement the frequency tuning step in SmartSDR, do the following:

1. Determine which Dial control you want to use. We'll use #103 in this example.
2. Using the **Edit Dial Setting** option on page 8 of the *Settings* menu, select the fourth button on the first menu (#103). Edit its label as needed, then touch the **Type** option to step to **Button**.
3. Open the **Edit Mapping** menu in the **CTR2 Dial Controller Tool** in SmartSDR
4. Select control 103 on the **Dial control** then turn the dial CCW. MIDI button 56 should be highlighted in the **Edit Mapping** menu in SmartSDR.
5. Select the *Tune Step Previous* function from the function list.
6. Turn the dial CW. MIDI button 57 should highlight in the **Edit Mapping** menu in SmartSDR.
7. Select the *Tune Step (1,20,100,1000)* function from the function list.

Now, turning the knob for Dial control 103 will increment or decrement the frequency tuning step in SmartSDR.

This is especially useful on **CTR2-Quad** with it's dual-function knobs. For instance, you can map the VFO control to knob D's turn function and the dial control set for the Button type (#103 in this example) as the push-and-turn function for knob D. Now, you can tune your radio with knob D and press and turn it to change the tuning resolution.

Home Page

The user interacts with CTR2-Dial using the touch screen to select from a series of menus and control pages. The ***Home*** page is the default display and is titled **CTR2-Dial**. From here you can select the various control pages and **Settings** menu.

There is a single physical button labeled **M5**, (yes, the label is upside down) built into the orange ring of the M5Dial. To press it, press the gray ring directly above the **M5** label. M5 is circled on the photo at the right.

The small triangle at the top of the display (just above **CTR2-Dial**) indicates the unit is using Bluetooth to connect to the app.

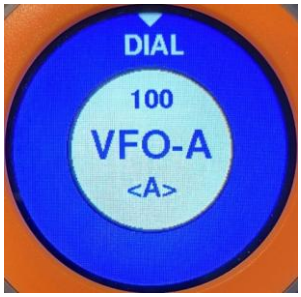
Press the **M5** button in any control or menu page to return to the ***Home*** page.

From the ***Home*** page you can quickly access the **Dial**, **Buttons**, **Knob**, and virtual **PTT** pages or open the **Settings** menu.

Pressing the **CTR2-Dial** caption at the top of the ***Home*** page will return you to the previously displayed menu or control.

Dial Control

The virtual *Dial* control supports 18 MIDI wheel or slider controls. The dial touch pad is the outer ring around the round center label. Swipe your finger around the outer ring to adjust the control.



Each dial control is assigned a MIDI control number starting at 100 and is shown of the top of the center label. This # indicates the control # you need to map in your app's MIDI function menu to execute this control.

The label for the selected control is shown below the MIDI control #. The label lets you know which function the app is *supposed to* execute for this control. If the label doesn't match the MIDI function mapped in the app, you can edit it to match.

- If the **Ring** option in the *Settings* menu is set for **Ring: Navigation** the gray ring on the M5Dial can be used for menu and control navigation. When used for navigation, turn the ring to select another dial control.
- If the **Ring** option in the *Settings* menu is set for **Ring: Fine-Tune**, touching the outside ring on the display enables *Ring fine-tuning mode*. In this mode, turning the ring will adjust the selected control by one count per ring click. Fine-turning will turn off after 3 seconds.

- If the **Ring** option in the *Settings* menu is set for **Ring: Sync2Dial**, disables ring navigation (switching pages or *Dial* controls with the ring) and adjusts the selected control in the *Dial* page as you turn the ring on any page. To navigate between pages, touch the page #s at the bottom of the display, or touch the control # in the *Dial* control to open the Dial menu
- If the **Ring** option in the *Settings* menu is set to a *Dial* control (for example **Ring: Volume**), ring navigation is disabled and turning the ring on any page adjusts the *Dial* control selected in the **Ring** option. To navigate pages, touch the page #s at the bottom of the display, or touch the control # in the *Dial* control to open the Dial menu.

Hint: Quickly change to the *Buttons* or *Knob menus* by swiping across the center of the display either *left* or *right*. Swipe from the *left* to open the *Knob menu* or from the *right* to open the *Buttons menu*.

Dial Graphics

The **Dial's** background depends on the MIDI *control type* selected for the control.

Control types can be set using a dumb terminal program, or you can use the **Edit Dial Setting** in the *Settings* menu.

The lead photo above shows a wheel control. Wheels are continuously variable and used for tuning. The MIDI control # (100) is shown on the top line of the center circle. The label in the middle (in this case **VFO-A**) indicates the control function and can be edited by the user. **<A>** at the bottom of the label indicates that this is a *WheelA* control.

In addition to wheel controls, you can select from two slider controls.

SliderA control is the most common. The outer touch ring on the **Dial** control represents a gauge with the level indicated by the filled arc that starts on the bottom left. In the photo to the right the volume is set at 26% and the level gage is 26% full. The MIDI control # (101) is displayed at the top of the center label.

SliderB is used for RIT, XIT, and other +/- offset functions. In this display the offset is represented by an arc that starts at the top center of the display. For negative offsets it grows to the left and for positive offsets it grows to the right from center. The MIDI control # (107) is displayed at the top of the center label.

When the RIT/XIT control is offset as shown, touching the center label (**RIT>**) resets the offset back to 0. You can enable/disable RIT or XIT from the *Buttons menu*. You can also assign the enable/disable RIT or XIT buttons to one of the knob switches.

You can also select buttons instead of wheels to be used with a *Dial* control. These are typically used in Marcus' apps that support button up/down controls for frequency control. Generally, you won't use buttons with the *Dial* but the option is there if you want to try it.

Touching the center label opens the *Dial menu* where you can select a control for the *dial control*. You can also rotate the gray ring on the M5Dial to change the selected dial control.

Shortcut: Touch to the *left* of the **DIAL** caption at the top to switch to the *Knobs menu* or touch to the *right* to switch to the *Buttons menu*. The *Knobs* and *Buttons menus* have the same controls.

Tuning Lock

To lock tuning to prevent inadvertent changes simply long-press the label on a control that is set for a Wheel MIDI type (typically VFO controls). When tuning is locked, the *Dial control* will be "grayed out" as shown here. Adjusting the control, or turning a knob assigned to this control will post a message with instructions on how to reset the lock.

To unlock the control, long-press the center label.

Ring Control Options

The encoder used in the gray ring control on the M5Dial display is a detent (bumpy) encoder with 12 "clicks" per rotation. Because it isn't very sensitive, and somewhat difficult to turn, it has only been used for selecting dial controls and menu pages (i.e. navigation).

By user request, beginning with v2.02 I've added the ability to choose from several ring control options with the **Settings** menu's **Ring** option on page 7. v2.04 adds the **Fine-Tune** option.

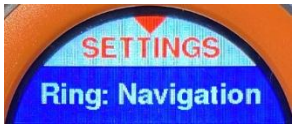
- **Navigation** mode – (default) – use the ring to select dial controls and menu pages.
- **Fine-Tune** mode – when enabled, touch the outer ring on the *Dial control* display to turn on fine-tuning. **<RING>** will be displayed in the page caption. Turning the gray ring will change the selected control's value by one count. **Fine-Tune** mode will time out after 3 seconds of inactivity.
- **Sync2Dial** mode – the ring adjusts the dial control selected in the *Dial control* (or on the *Dial menu*) – this works in any page, similar to how the knobs work.
- **Selected Dial Control** – the ring adjusts a dial control that you select from the **Ring Ctl** menu in **Ring:** option in the *Settings* menu. This also works from any page.

The selected ring mode is indicated by vertical lines the Bluetooth antenna icon area at the top of each page. No lines indicates the ring control is in **Navigation** mode. A single line indicates the ring is in **Fine-Tune** mode. Two lines indicates **Sync2Dial** ring mode is active. Three lines indicates a selected *dial control* is assigned to the ring control.

NOTE: When **Navigation** or **Fine-Tune** modes are not selected you must use the on-screen navigation controls to move between menu pages and select **Dial controls**.

The following screenshots should help explain the process of selecting a ring control option.

Navigation mode



The screenshot at the right shows that the ring is in **Navigation** mode.

In this mode, turning the ring will change menu pages or the selected dial control.

No lines are shown at the top of the display.

Touch the **Ring: Navigation** button to switch to **Fine-Tune** mode.

Fine-Tune mode

In this mode the ring can be used to fine-tune the selected control while in the *Dial control* page. Touch the outside ring on the *dial control* to enable fine-tuning. This mode will timeout after 3 seconds of inactivity.

A single line in the Bluetooth antenna area is visible in this mode.

Touch the **Ring: Fine-Tune** button to switch to **Sync2Dial** mode.

Sync2Dial mode

In **Sync2Dial** mode the ring controls the currently selected **Dial control** from any page, similar to how the **Knob** controls work.

Two lines are shown in the Bluetooth antenna icon area when this mode is active.

Touch the **Ring: Sync2Dial** button to open the **Ring Ctl** menu.

Ring Ctl Menu

The *Ring Ctl* menu appears when you touch the **Ring: Sync2Dial** button to change the ring mode. This menu is similar to the *Dial menu* and is used to assign one of the dial controls to the *Ring* control. There are three pages of menus. Use the **Next** button on the bottom to step to the next page. Touch a control label to assign that control to the **Ring** control. In this example, the **AGC-T** control is selected.

Pressing **Esc** in this menu returns the **Ring** mode back to **Navigation** mode without selecting a control.

Selected Dial Control Mode

Once you select a control in the **Ring Ctl** menu you will return to the **Settings** menu and the **Ring:** option will show the selected control.

Note the three lines in the antenna icon area at the top of the page.

To return to ring **Navigation** mode, touch the **Ring:** button again.

Dial Menu

If the gray ring on the M5Dial is set for navigation or fine-tune mode, it can be used to change the *Dial control's* function. Otherwise, to change the virtual dial's function you'll need to open the *Dial menu*. To do this, touch the MIDI control# on the *Dial control*. This menu allows you to select from a three-page menu of dial functions. The current dial setting will be highlighted (**VFO-A** in this example). Touch the left or right side of the bottom button or twist the gray ring on the M5Dial (if in ring navigation or fine-tune mode) to move to the next page.

Touch a button to select that control and return to the virtual dial control, or touch the selected function to return to the virtual *Dial control* without changing the selection.

The caption button (**DIAL MNU**) is a two-function button.

- Touch the left side to switch to the **Knob** menu.
- Touch the right side to switch to the **Buttons** menu.
- To prevent unintended operation, the caption button requires one touch to select the function and one touch to execute the function.

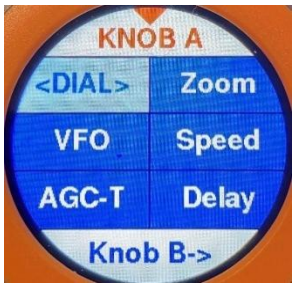
Hint: Quickly change to the *Knob* or *Button* menus by swiping across the center of the display either *left* or *right*. Swipe from the *left* to open the *Knob menu* or from the *right* to open the *Buttons menu*.

Knob Menu

The *Knob menu* format depends on the hardware **CTR2-Dial** is running on.

CTR2-Uno and CTR2-Duo Controllers

When running on these hardware platforms (or on the older CTR2-Flex and CTR2-Dial Hardware), the **Knob** menu presents six wheel/slider functions that you can assign to a knob. On dual encoder models each knob can be assigned to one of six wheel/slider functions.



On the photo at the right, the first option (**<DIAL>**) is selected. This is a special function that sets *Knob A's* function to the current *Dial control's* selection.

If the VFO function is selected on the *Dial control*, turning *Knob A* while on any page will change the VFO frequency. If the *Dial control* is changed to Volume, turning *Knob A* will change the volume.

Selecting one of the other five functions on this page assigns it to *Knob A.* These five functions can be edited to fit your operating preferences.

On the dual-knob model shown, each knob has its own menu page with six functions. On single knob models have one page of six functions.

CTR2-Quad Hardware

The *Knob* menu on the **CTR2-Quad** is different than the menu for the **Uno** and **Duo** controllers, but editing the selected controls is the same as with the other units.

CTR2-Quad supports dual-function knobs. Each knob has a *turn* and a *push-an-turn* function.

The menu you are in is displayed in a label on the left side of the menu. In this photo, we're in the *Set Knob Turn Function* menu.

The functions assigned to each knob are show on the buttons.

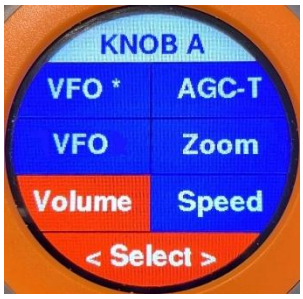
The *Knob* menu is laid out the way the knobs are positioned on the **Quad**. *Knob A* is the bottom-left button (and shown selected in this photo), *knob B* is the upper-right button, *knob C* is the right-middle, and *knob D* is the bottom-right button.

Touching the black label toggles the menu to the *Set Knob P&T Func* (*Push-and-Turn*) menu. This menu allows you to assign the *Dial control* that is used when you push the knob down and turn it (while holding it down).

NOTE: Any knob on **CTR2-Quad** can be assigned to the **<DIAL>** function. When this function is selected, that knob will control the function selected in the *Dial control*.

Editing the Knob Menu

You can easily change the functions available in the *Knob* menus. Just touch a function to select it, then touch it again to edit it. The selected function and the bottom button will turn red. Turn *Knob A* (or *Knob B* on a dual-knob unit) to scroll through the functions available in the *Dial menu*.



NOTE: Unlike the *Dial* and *Button* controls, you can edit the MIDI command # assigned to each control in the **Knob** menu using a terminal program. This allows you to map MIDI commands that are not hardcoded to the other controls.

Touch the **< Select >** button on the bottom or let the select timer timeout (3 seconds) to save.

To prevent unintended operation, the caption button (**KNOB A**) is a dual-function button and requires a select and execute touch.

- Touch the *left* side to switch to the *Dial control*.
- Touch the *right* side to switch to the *Buttons Menu.*

Touch the bottom button (**Knob B->** or **Page 2->**) or turn the gray ring on the M5Dial (if the ring is in navigation or fine-tune mode) to change the *Knob menu* page.

Hint: Quickly change to the *Buttons* menu or *Dial control* page by swiping across the center of the display either *left* or *right*. Swipe from the *left* to open the *Buttons* menu or from the *right* to open the *Dial control* page.

Buttons Menu

Open the *Buttons menu* from the *Home* menu to select from 42 button functions. Touching a button immediately sends the MIDI control for that function.

There are seven pages of buttons with six buttons on each page. If ring navigation or fine-tune mode is enabled, turn the gray ring on the M5Dial to select a page, otherwise, pressing the *left* or *right* side of the button at the bottom of the display (**1/7** in this photo) to decrement or increment the page.

To prevent unintended operation, the caption and footer buttons are dual-function buttons. The first touch selects the button and the second touch executes the selected function.

- On the caption button touch the left side to switch to the **Knob Menu** page. Touch the right side to switch to the **Dial** page.
- On the footer button touch the left side to decrement the menu page number. Touch the right side to increment the menu page number.

Press the **M5** button on the orange ring to return to the ***Home*** page.

Hint: Quickly change to the *Dial control* or *Knob menu* by swiping across the center of the display either *left* or *right*. Swipe from the *left* to open the *Dial control* or from the *right* to open the *Knob menu*.

Assigning a Knob Switch to a Button Function

Knob switches (and pushbuttons on a **CTR2-Quad**) can be set to two actions: a short-press (less than 1 second) and a long-press (greater than 1 second).

Switch Short-Press Action

To assign a short-press button action to a knob's push switch (or pushbutton on a **CTR2-Quad**), do the following:

- Open the *Buttons* menu and navigate to the page with the function you want to assign
- Press and Hold the *Knob* (or *pushbutton* on the **Quad**) switch you want to use
- Touch and release the function button you want the switch to execute

Once you have assigned a function to a *Knob* (or *pushbutton* on the **Quad**), the knob's letter (or switch #) will appear in that button's label. That function will be sent to the radio anytime you press that *Knob* (or *pushbutton*).

Switch Long-Press Action

To assign a long-press button action to a knob's push switch (or pushbutton on a **Quad**), follow the steps above for setting the short-press action then while still holding the knob (or switch) down, touch the

desired function a second time. *Buttons* set to long-press actions will have a **~** appended to the knob (or switch #) displayed on the button.

The photo above shows the *Buttons* menu on a **CTR2-Quad.** *Knob A's* short-press action is set to the **Mute** function and its long-press action is set to the **Tune ^** function. *Knob D's* short-press function is set to **Band ^** and its long-press action is set to **Band v**.

To clear a function, hold the **Knob** switch down and touch the selected function again.

NOTE: You must clear both switch actions from the button before you can assign it to another knob or switch#.

NOTE: Changing the PTT assignment on **CTR2-Quad's** pushbutton #1 or #2 automatically cancels any switch action assigned to those switches.

NOTE: *Buttons* are not set in the factory default settings. You will always need to assign the knob and switch actions to fit you operating preferences.

PTT Page

The **PTT** page offers a virtual PTT button. Simply touch the display to key your radio and lift your finger to unkey. The MIDI control # sent to the app is determined by the **PTT Ctrl #** selected on page 5 in the **Settings** menu.

If your radio control app supports latching PTT, map to that function in your app to the MIDI control # selected for PTT in the settings menu. In latching mode, touching and releasing the display keys PTT. Touch and release the display again to unkey PTT.

NOTE: If your app supports latched PTT, the display on your **CTR2** unit will not remain red when you lift your finger after key PTT.

The default MIDI control # for PTT is 99. You can choose from 96, 97, 98, or 99 in the **Settings** menu under **PTT Ctrl**.

NOTE: Controls 20, 21, 30, and 31 are also available in **PTT Ctrl** to allow compatibility with apps such as RHR and RemoteTx that are configured for **CTR2-MIDI**. This will be removed if RHR and RemoteTx updates their support to include **CTR2-Dial**.

NOTE: If your radio keys up when you press the PTT display but your microphone doesn't produce power output, check the microphone settings in the app.

Using a Hardware PTT Switch and Straight Key

You can use a straight key (or external keyer) and/or an external PTT switch connected to the **Paddle Input** jack on your **CTR2** unit. To do this:

1. Connect your straight key (or external keyer) to the TIP and SHIELD (ground) terminals on a 3.5mm (1/8") stereo plug. Connect your PTT switch between the RING and SHIELD on the same plug.
2. Plug the stereo plug into the *Paddle Input* jack on **CTR2-Uno**, **CTR2-Duo** (or the older **CTR2-Dial** and **CTR2-Flex** hardware). On **CTR2-Quad**, plug it into the *Key/PTT* jack.
3. In the *Settings* menu, set **Pdl->** to **MIDI: 98/99** (or **30/31** if using Remote Ham Radio or RemoteTx). Contact closures on the input jack will send MIDI commands 98 and 99 (or 30 and 31).
4. In the **Settings** menu, set **PTT->** to **MIDI: 99** (or **31** if using CTR2-Dial with Remote Ham Radio)

Once configured, pressing the straight key will send MIDI command 98 (or 30) and pressing the PTT switch will send the MIDI control defined in **PTT Ctrl** to your app to key PTT. Make sure those control # are mapped to MOX or PTT in your app.

Settings Menu

To enter the *Settings menu*, press the **Settings** button at the bottom of the ***Home*** page.

The settings menu is divided up into nine pages with three buttons on each page. You must press a button twice to change its setting.

Press the **M5** button or touch the **SETTINGS** caption to return to the ***Home*** page**.**

To change menu pages touch on the left or right side if the footer button (**<8/9>** in this example) or turn the gray ring if ring navigation or fine-tuning mode is enabled.

The table below summarizes each setting option.

Page	Btn	Function	Description
1	1	**Beep Mode**	Changes the volume of control beeps: Off, Low, Medium, and High
	2	**Speed Tune**	When enabled, touch and hold the outer ring on the **Dial** for > 1 second to enter speed tuning mode. Valid for *WheelA* and *WheelB*.
	3	**Dial Sens**	Changes the sensitivity of the virtual dial. Range: Low, Medium, High. Default is Medium.
2	1	**Ctrl Map**	Selects control Map 1 or 2. Each map can be configured for a specific radio control app. Use a different color theme for each map. Default is *Map 1*.

2	2	**Theme**	Select the color theme for the selected Map. Options include Dark, Light, Blue, Orange, Green, Red, and Violet. Default is Blue for Map 1 and Violet for Map 2.
	3	**Backlight**	Adjusts the backlight. Range: Low, Medium1, Medium2, and High. Lower settings reduce current draw on the unit.
3	1	**Sleep**	Sets a timer that turns off the display backlight after no activity for the set number of minutes to reduce current draw. Sleep mode does not close USB or Bluetooth connections. Range: Never, 1, 5, 10, 30, and 60 minutes.
	2	**Pwr Off**	Sets a timer to turn off the unit after no activity for the set number of minutes. Pwr Off mode reduces current draw to 14 microamps and closes USB and Bluetooth connections. Range: Never, 10, 30, 60, 120, and 180 minutes.
	3	**Power Down**	Immediately closes the Bluetooth and USB connection and puts the unit to sleep, reducing current draw to 14 microamps. Touch the display to wake up the unit. **NOTE:** If you are powering your **CTR2** controller with a 3.7-volt battery you will need to press and hold the **M5** button on the orange ring for > 10 seconds to wake the controller up from sleep mode.
4	1	**# Knobs**	Selects the number of external knobs your unit is equipped with. (One and two knob configurations cannot be detected in the firmware.) For **CTR2-Uno**, **CTR2-Duo**, (and older **CTR2-Flex** and **CTR2-Dial** hardware) you have the following options: **None**, **One, Two**, or **Swap A/B** (for two knob units) • When this firmware is running on **CTR2-Dial** hardware the paddle jack is only available in the **None** and **One** options. **CTR2-Quad** hardware is automatically detected and this setting is fixed to **# Knobs: Four**
	2	**Knob/Knob A Or SW1 Setup**	On **CTR2-Uno**, **CTR2-Duo**, (and older **CTR2-Flex** and **CTR2-Dial** hardware) this option selects either Normal or Reverse direction for *Knob A*. The default is **Normal**. On **CTR2-Quad** this option toggles the PTT setting of pushbutton #1. You can use this switch instead of the virtual PTT page or an external PTT switch to key your radio. **NOTE:** Toggling this setting resets any button switch actions that were assigned to SW1 in the *Buttons menu.*
	3	**Knob B Or SW2 Setup**	On **CTR2-Duo**, (and older **CTR2-Flex** and **CTR2-Dial** dual-knob hardware) this option selects either Normal or Reverse direction for *Knob B*. The default is **Normal**. This option is not available on single knob units. On **CTR2-Quad** this option toggles the PTT setting of pushbutton #2. When PTT is enabled, you can use this switch instead of the virtual PTT page or an external PTT switch to key your radio.

4	3		**NOTE:** Toggling this setting resets any button switch actions that were assigned to SW2 in the *Buttons menu.*
5	1	**Pdl->**	Selects the MIDI control #s mapped to the *Paddle Input* jack. Default is 96/97 and these are usually mapped to the *Left* and *Right* paddle controls in the app. Option 98/99 can be selected and mapped to the Straight Key and PTT options in the app. For compatibility with Remote Ham Radio and RemoteTx's mapping for **CTR2-MIDI** you can also select 20/21 and 30/31 pairing. **NOTE:** By default, controls 20/21 and 30/31 are mapped to buttons. Remove these mappings in your app when controlling them with the paddle input.
	2	**PTT->**	Selects which MIDI control # is mapped to the virtual **PTT** button page and the PTT input jack, and the switches on **CTR2-Quad**. The default 99 is typically mapped to PTT or MOX in the app. Range: 96, 97, 98, and 99. MIDI controls 20, 21, 30, and 31 are also available for compatibility with Remote Ham Radio and RemoteTx's **CTR2-MIDI** mapping. To use an external PTT switch, wire it to the RING and SHIELD of the **Paddle Input** jack and select **98/99** on the **Pdl In** option above and set **PTT Ctrl** to **99**. If using Remote Ham Radio or RemoteTx, select **30/31** on the **Pdl In** option and set **PTT Ctrl** to **31**. See Using a Hardware PTT Switch and Straight key.
	3	**PDLS or PDLS & KEY**	Swaps the TIP and RING inputs on the **Paddle Input** jack on **CTR2-Uno** and **CTR2-Duo** (and older **CTR2-Dial** and **CTR2-Flex**) hardware. On **CTR2-Quad** hardware, this option swaps the TIP and RING on either the *Paddle* input jack and/or the *Key/PTT* input jack.
6	1	**Import Settings**	Import settings from a previously exported setting file using XModem. In Tera Term select *File->Transfer->XMODEM->Send* and select a previously saved settings file on your computer. Next, touch **Import Settings** to start the transfer.
	2	**Export Settings**	Backup **CTR2-Dial's** settings by exporting them to a file on your computer. To do this, in Tera Term select *File->Transfer->XMODEM->Receive* then enter the name of the file you wish to save. Next, touch **Export Settings** to start the transfer. Once this file has been saved you can recover from a unintended memory erase when flashing new firmware by using the **Import Settings** option above. **NOTE:** You can also use the import/export options to create setting files for different radio control apps.
	3	**Info**	Displays program version #, registered call sign, registration key, and Bluetooth ID
7	1	**Ring Mode**	Selects the operating mode of the gray ring on the display. See Ring Control Options for a complete discussion on Ring modes.
	2	**Rotate Screen**	Rotates the screen 90 degrees for each selection. This allows the screen to be aligned for various M5Dial mounting configurations
	3	**Touch Delay**	Enables or disables a 120-millisecond delay on touch input. This can be used to reduce unintended touch events and occasional randomly generated touch events from the M5Dial touch sensor.

8	1	**Edit Dial Setting**	Opens the Dial control label and MIDI control type editor. See Editing CTR2-Dial's Labels and Control Types for more information on dial labels and MIDI control #s.
	2	**Edit Btn Label**	Opens the Button label editor. See Editing CTR2-Dial's Labels and Control Types for more information on button labels.
	3	**BT**	Turns the Bluetooth radio on or off. When using a USB connection to your radio control app you can turn the Bluetooth radio off and save about 50 milliamps of current draw.
9	1	**Run CTR2-Flex**	Available on v2.xx firmware. Shuts down the **CTR2-Dial** MIDI controller firmware and restarts the unit with **CTR2-Flex** WiFi controller firmware.
	2	**Restore to Start Up**	This option allows you to reset your settings to what they were when you first booted **CTR2-Dial**, *if you didn't choose to reset your call and registration key in* ***Reset to Factory***. This is handy in case you want to "roll back" changes you made during the current session.
	3	**Demo Mode: Edit Call & Key** **Normal Mode: Reset to Factory**	**In Demo Mode:** This option allows you to enter your call and registration key without using a terminal program. To enter text, use the ring encoder on the M5Dial or Knob A to select a character and touch the display or press and turn Knob A to move to the next character. Press **OK** to accept. You can also connect a terminal program and use the terminal keyboard to enter your call and registration key. **In Normal Mode:** This option resets CTR2-Flex's configuration back to the original factory settings. **NOTE:** You will be given the option to reset your call sign and firmware registration key. **NOTE:** If you reset the configuration by mistake, and didn't choose to reset your call and registration key, execute the **Restore to Start Up** option to restore your settings before powering the unit down.

Label and Settings Maps

CTR2-Dial supports two maps for control settings. Select the map you want to use on page 2 of the *Settings* menu. These maps store control labels and MIDI control types for the dial controls. They do not change the MIDI command #s assigned to the controls in **CTR2-Dial**. You must *edit your apps' map* to change the *function* the app executes for MIDI command #'s that **CTR2-Dial** sends.

You can edit the labels and control types in each map to fit your needs. For instance, you can configure Map 1 to contain the labels and control types for SmartSDR and Map 2 for labels and control types for SDR-Control. Simply switch maps to match the app you're using.

HINT: I recommend using different color themes on each map so it's easy to determine which map you're using.

Editing Labels and Control Types

Prior to v2.02.00 firmware, to edit the labels or MIDI control types associated with each control in **CTR2-Dial** you had to run a dumb terminal program such as Putty or Tera Term. Refer to Appendix C, D, or E for information on connecting to a terminal.

v2.02.00 adds additional editing options. You can now choose from two new control editors that allow you to edit the properties associated with **CTR2-Dial's** *Dial* and *Button* controls without using a terminal.

The Dial editor allows you to edit the label and the MIDI control type assigned to each *Dial control*.

The Button editor allows you to edit the label associated with each *Button* control.

These editors can be found on page 8 in the *Settings menu*. For more information on how **CTR2-Dial** controls your radio control app, see How CTR2-Dial Works with Your Control App.

Dial Editor

When you select the **Edit Dial Settings** option the **EDIT DIAL** menu opens. This menu is similar to the **Dial menu** in that it has three pages of dial controls to choose from. The selected dial control will be highlighted, but it's not used in this menu.

Turn the gray ring on the M5Dial (if ring navigation or fine-tune is enabled), or touch the left or right side of the bottom button (**< 1/3 >**) to select a menu page.

Touch the label of the dial control you want to edit. The edit page will open. Touch the **EDIT DIAL** caption to cancel this mode and exit back to the *Settings* menu.

The MIDI control # for the selected dial control will be displayed in the caption of the edit page. MIDI control # is hardcoded and cannot be edited.

The edit cursor will highlight the character to be edited ("V" in this example). Turn *Knob A* to change the character. Touch **Next** to move the edit cursor to the next position, or touch another character in the label to move the edit cursor to that position. There are eight characters available for labels.

NOTE: If you are running **CTR2-Dial** firmware on **CTR2-Uno**, **CTR2-Duo**, **CTR2-Quad**, or older **CTR2-Flex** hardware, you can press and turn *Knob A* to move the edit cursor.

Touch **Ins** to insert a character at the cursor position and **Del** to delete the character at that position.

To change the MIDI control type associated with this control, touch **Type:** on the display. The MIDI control type will step to the next control type. Refer to the MIDI Control Types section for a description of each control type.

When you're done editing, touch **OK**. To exit without saving your changes, touch **Esc**.

Button Editor

When you touch the **Edit Button Labels** the **EDIT BTN** menu opens. This menu has 7 pages of buttons with six buttons on each page, similar to the *Buttons menu*. Turn the M5Dial's ring if ring navigation or fine-tune mode is enabled, or touch the left or right side of the bottom button (**< 1/7 >**) to move to the next page. Touch the **EDIT BTN** caption to cancel this mode and exit back to the *Settings* menu.

Once you select a button to edit the **Button Editor** opens. The MIDI control # is shown in the caption – it cannot be changed.

The **Button Editor** has the same features as the **Dial Editor** but it doesn't have a MIDI **Type** setting. This is because buttons always send MIDI button On/Off commands.

Editing With a Terminal

Onscreen editors are useful for small editing changes but if you have several controls to edit, it's easier to use a dumb terminal.

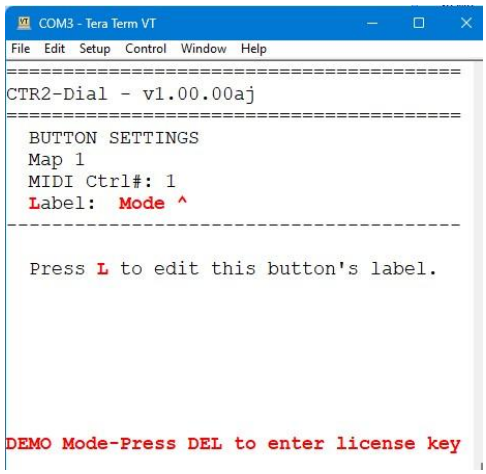
Start your terminal program and connect it to the virtual USB serial port assigned to **CTR2-Dial** by your operating system.

When the terminal first starts you will be presented with a blank screen because the terminal server is waiting for you to wake it up. This is done by pressing any key (usually [Space]) or by selecting a control on the display.

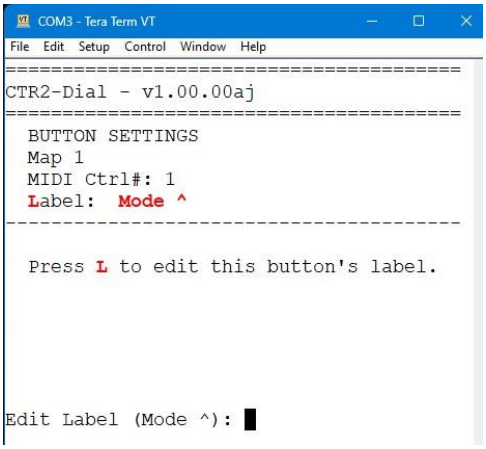
NOTE: Dial controls without labels will not show up on the *Dial Control* unless a terminal is connected to **CTR2-Dial**. To add a label to a blank control, select it in the *Dial menu*.

The header of each page contains the firmware version #.

If you provide the M5Dial for your unit, the first time you connect, **CTR2-Dial** will be running in **Demo** mode and a note will be displayed at the bottom of the terminal page instructing you to press the **DEL** key to enter your call and registration key. If you don't have a key, you can ignore this message and remain in **Demo** mode. In this mode the program will shut down in 15 minutes and changes you make will not be saved. Other than that, the program is fully operational.



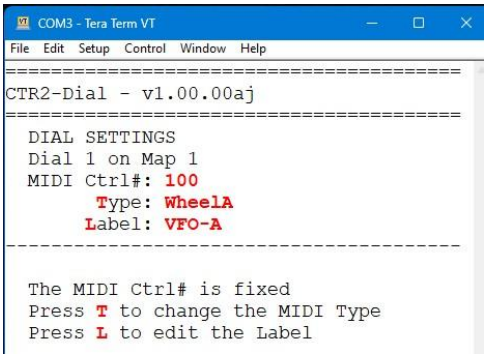
To edit a control, simply select it on the **CTR2** unit. The terminal page will update with that control's settings. In the screenshot at the right, **CTR2-Dial** is on page 1 of the *Button menu*. The first button on the selected page is always displayed first. To edit the label on this button, press the **L** key on the terminal. The **Edit Label:** line will appear with the current label in parenthesis. Enter the new label at the prompt.



Wheel and slider controls on the *Dial* and *Knob* pages include additional settings.

Dial controls are assigned to MIDI control #s 100 through 117. You cannot change these. You can edit the MIDI control # assigned to *Knob* controls. This allows you to send MIDI control #s that aren't included in the default settings.

Press **T** to step through the MIDI control type assigned to the control or press **L** to edit the label.



Once you have **CTR2-Dial's** map configured the way you want, edit the map in your radio control app to match.

You can export your settings to your computer so you have a backup copy. This process will be described next. You can also import settings from other configurations. You can find configurations on the Lynovation web site on the CTR2 Configuration Files page.

NOTE: Backup files created with v1 firmware are not compatible with v2 firmware.

Backing Up and Restoring your Settings

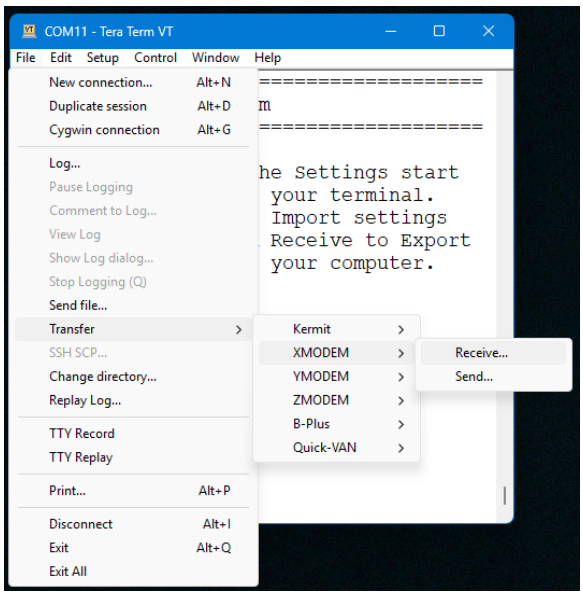
Once you have your **CTR2-Dial** setup the way you want it, go to the **Settings** page and **Export** your settings to back up the file on your computer.

CTR2-Dial uses the XModem file transfer protocol to transport settings to and from backup files on your computer. XModem is an older protocol and not many terminal programs support it anymore. Tera Term does and I'll use it in the document. Putty does not. To get XModem functionality with Putty you need to download and install ExtraPutty. This program is a branch off of Putty. You can download it from here.

Before you can use the **Import Settings** option you must export the settings from **CTR2-Dial** once you've made changes. If you want to just return to factory settings, select the **Reset to FACTORY** option in the *Settings menu*.

Export Settings

To export the current settings from **CTR2-Dial**, connect your dumb terminal to **CTR2-Dial** then open its XModem transfer utility. On Tera Term, open the *Files->Transfer->XMODEM->Receive* option shown here.



When you select this option Tera Term will ask you to enter the file name for the file you want to create. Navigate to the folder you want to save it in then give it a name. I prefer to use the *.txt* extension but you can use any extension you want.

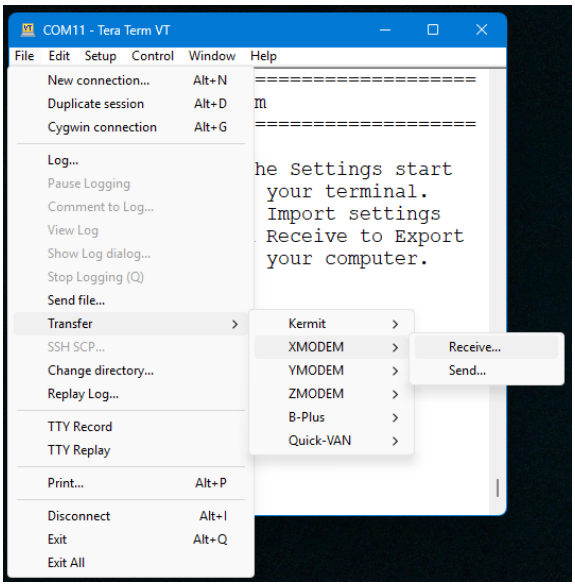
NOTE: If you don't enter an extension, your file will be created without one. This can cause Windows a little heartburn.

When you click OK, Tera Term will open the XModem transfer window and wait for the transfer to start. At this point, touch the **Export Settings** option in the **Settings** menu. The transfer should complete in a second or two and you should have a new file in your backup folder. You can view this file with a text

editor. I don't recommend editing the values in this file because invalid values may cause problems with **CTR2-Dial's** boot process.

Import Settings

Now that you have a backup file of your settings (or have downloaded a configurate file from the Lynovation web site) you can import them here.



To import a settings file in Tera Term, navigate to the *File->Transfer->XMODEM-Send* option. Once there, Tera Term will ask you to select a file from your backup directory. Select one of the files you previously saved using the **Export Settings** option. Tera Term will then open the file transfer window and wait for **CTR2-Dial** to tell it to start. Touch the **Import Setting** button to start the transfer. It should only take a second or two. Once it completes **CTR2-Dial** will load the new configuration.

HINT: You can use the export and import settings functions to save and reload multiple versions of your settings. Perhaps you use **CTR2-Dial** on more than two apps and have each app configured differently. You can export the settings for each of your apps then import them back in when you want to change apps. This is also handy if you have more than one **CTR2** controller, or if you want to share your configuration with other users.

Control App Maps

The following sections describe how to map several apps that you may use. It's not an inclusive list of all the apps that support MIDI control. These instructions are included in the configuration files available on the Lynovation web site under CTR2 Configuration Files.

Before editing the settings in **CTR2-Dial**, you may want to export the settings to your computer in case you need to restore them in the future.

Remote Ham Radio Maps

The current release of Remote Ham Radio does not include mapping for **CTR2-Dial**. This may change once **CTR2-Dial** gains popularity on the platform. The instructions below allow you to change the map in **CTR2-Dial** to fit the current *MIDI* mapping in RHR.

In the RHR console, open the **Tools** menu and select **CTR2-MIDI**.

Connect a terminal to **CTR2-Dial** and change the labels on the following controls:

Keying Input

Navigate to page 5 in the *Settings menu* and change the **Pdl->MIDI:20/21** to use your paddles to control the keyer in RHR. Change **Pdl->MIDI:30/31** to use the paddle input jack to control the straight key and PTT functions instead.

Dial Controls

The **Dial** labels should be edited in a terminal or with the **Edit Dial Setting** option in the *Settings menu* to match the function assigned in RHR.

DIAL ACTIONS		
Dial #	**MIDI Control #**	**Action Taken**
1	100	Change VFO
2	101	Slice Volume
3	102	NR Level
4	103	NB Level
5	104	VOX Level
7	106	Keyer Speed

The other **Dial** controls are not currently used. To avoid confusion set their labels to a single space. This will hide them on the *Dial control* page (but not in the *Dial menu*) until you connect a terminal to **CTR2-Dial**.

Button Controls

The number of buttons available in **CTR2-Dial** has been greatly expanded over those available on the **CTR2-MIDI**. This means that there will be a lot of holes in the *Buttons menu* in **CTR2-Dial** when set to

match the current mapping used in RHR for the *MIDI*. Future versions of RHR will hopefully consolidate the buttons and may add more functions.

BUTTON ACTIONS			
Button and MIDI Ctrl #	**Original *MIDI* Button**	**Action Taken**	**Dial Label**
1	MFB1	CW Mode	CW
11	MFB1 (long-press)	Toggle NR	NR
2	MFB2	SSB Mode	SSB
Button and MIDI Ctrl #	**Original *MIDI* Button**	**Action Taken**	**Dial Label**
12	MFB2 (long-press)	Toggle NB	NB
3	MFB3	FT8	FT8
13	MFB3 (long-press	FT4	FT4
4	MFB4	Toggle PTT	PTT
14	MFB4 (long-press)	Toggle VOX	VOX
5	MFB5	Band Down	Band v
6	MFB6	Band Up	Band ^

All other button labels on **CTR2-Dial** can be changed to a single space.

RemoteTx Mapping

Important: HID and RemoteTx support does not work in **CTR2-Flex** firmware.

To use **CTR2-Dial** with RemoteTx, click **VFO** in **RemoteTx's** *Radio* page, before you connect to the radio.

A list of USB devices on your computer will open. Select the **M5STACK_DIAL** device, then press **Connect**. The **VFO** button should turn red, indicating that the **CTR2-Dial** is connected to this browser session.

To use **CTR2-Dial** with **RemoteTx**, you'll want to edit the labels and knob control types to match the HID controls **RemoteTx** is programmed for. A spreadsheet showing the required mapping can be found at **CTR2 Configuration Files** on Lynovation.com. Download the zip file and navigate to the **RemoteTx** folder. This folder also contains a **CTR2-Dial** configuration file, *CTR2-Dial_RemoteTx.txt*, that you can import into your **CTR2** unit to change all of the settings for **RemoteTx**.

PTT Input

RemoteTx only supports PTT control. It does not support a CW keyer. Plug a straight key or PTT switch into the 3.5mm paddle jack on your **CTR2** unit. Wire your switch to TIP or RING and SHIELD of a 3.5mm stereo plug.

NOTE: RemoteTx is looking for MIDI control 20 or 21 for PTT. To set your unit up for this, open the *Settings* menu and navigate to page 5. Touch the **Pdl->MIDI** button until it shows **Pdl->MIDI: 20/21**. If you've wired your switch to the TIP and SHIELD of the paddle plug, touch the **PTT->MIDI** button until it shows **PTT->MIDI: 20**. If you wired your switch to the RING and SHIELD of the paddle plug, select **PTT->MIDI: 21**.

iOS and MacOS App Maps

Maps for Marcus' apps are available from the CTR2 group or from the **CTR2 Configuration Files** page on Lynovation.com.

To change the default map that comes with you iOS/MacOS app, first, make sure your app has permission to use your iCloud drive.

Load the app map you want to use into your iCloud folder. Next, open the **CTR2 Controller** object in the app's **Tools** menu then go to the **Edit Mapping** page. On this page, click or touch the **File** icon at the top of the page then select *Import Mapping from iCloud Drive*. This will open a list of files available on your iCloud drive. Select the appropriate map from the list.

<<< IMPORTANT >>>
Once you have loaded the map you must change one of the control assignments before the map will be saved to your app. To do this, select one of the controls, change it to something else, then change it back to its original setting. If you don't do this the map will revert back to the original map.

The default configuration for **CTR2-Dial's Map 1** is setup for SmartSDR for iOS/MacOS, *but it does not match the default map shipped with SmartSDR*. You will need edit this map manually or import the correct map from the Lynovation web site for your app. If you use other apps from Marcus (SDR-Control, FT-Control, TS-Control, or K4-Control) you'll also need to import the correct map to your app *and import the correct **CTR2-Dial** configuration file, or edit the labels in **CTR2-Dial** manually*. I have a video on YouTube explaining the mapping process.

Using Paddles with the App's Keyer

Marcus' iOS/MacOS apps include a keyer that you can control with the paddles connected to **CTR2-Dial**.

> **NOTE 1:** If you're using Bluetooth-LE MIDI you may notice a difference in how the paddles work with the app's keyer as opposed to a hardwired paddle/keyer. The latency in the Bluetooth-LE connection can cause up to 15 milliseconds of delay. This will throw your "fist" timing off, especially in Iambic B mode where an extra element may be added. It may take some practice to get use to the timing change. If this is a problem for you consider using USB MIDI instead.
>
> **NOTE 2**: Using Bluetooth headphones adds additional latency. You might consider using an adapter on you iOS device that supports a wired headphone along with an OTG connection to minimize this latency.

Follow these steps to configure the app and CTR2-Dial for paddle keying:

- In the app, map **Button 96** to **Trigger CW Left Paddle** and **Button 97** to **Trigger CW Right Panel**. You can swap this mapping if they are wired backwards.
- Plug your paddles into the **Paddle In** jack on your **CTR2** unit.
- Set the radio's mode to CW
- In SmartSDR, press the **View** menu in the Panadapter display and select **CWX Panel**. You can adjust speed and other setting here.
- In SDR-Control, select the **CW Keyer** option in the **Tools** menu to configure the keyer. DO NOT enable the MIDI device in this panel, this is done in the **CTR2 Dial Controller Tool** panel.
- Pressing either paddle will now activate the keyer and key the radio.

Using a Straight Key or an External PTT Switch

Straight Key Wiring

If you want to use a straight key with **CTR2-Dial**, map **Button 98** to **Trigger CW Straight Key** in the app then wire your straight key to the TIP and SHIELD of a 3.5mm (1/8") stereo jack and insert it into the **Paddle In** jack on your **CTR2** unit. Do not use a mono jack as the shield will ground the RING lead on the **Paddle In** jack.

PTT Switch Wiring

If you want to use an external PTT switch with **CTR2-Dial**, map **Button 99** to **PTT Push** in the app and wire your PTT switch to the RING and SHIELD of a 3.5mm (1/8") stereo jack and insert it into the **Paddle In** jack on your **CTR2** unit.

NOTE: You can wire your straight key and external PTT switch to the same stereo plug.

Select **Pdl In: 98/99** in page 5 of the **Settings** menu to tell **CTR2-Dial** to route the paddle jack inputs to MIDI control #s 98 and 99.

Thetis Maps

Thetis is a little different than RHR and iOS/MacOS apps in that it uses **CTR2-Dial's WheelB** control instead of **WheelA** for frequency tuning. **Map 2** is pre-configured with **WheelB** for tuning. You can select **Ctrl Map: 2** on page 2 of the *Settings menu*.

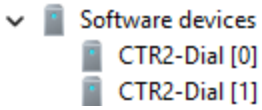
With the **Setup -> Setup -> Serial/Network/Midi CAT** window open in Thetis, select the **MIDI** tab and click the **Configure MIDI** button. The program will find and initialize **CTR2-Dial** using USB.

Windows doesn't support Bluetooth LE, so you must use USB with Thetis.

NOTE: If the program doesn't initialize **CTR2-Dial**, check to make sure you are using a USB-C data cable, not a USB-C charge cable.

It's a Software Device

If you can't get Thetis to recognize **CTR2-Dial**, verify that **CTR2-Dial** appears in the **Device Manager** under **Software devices**, and not under **Sound, Video, and game controllers**. If it appears here, right click on the entry and uninstall it. Cycle the power on you PC and try connecting **CTR2-Dial** again.



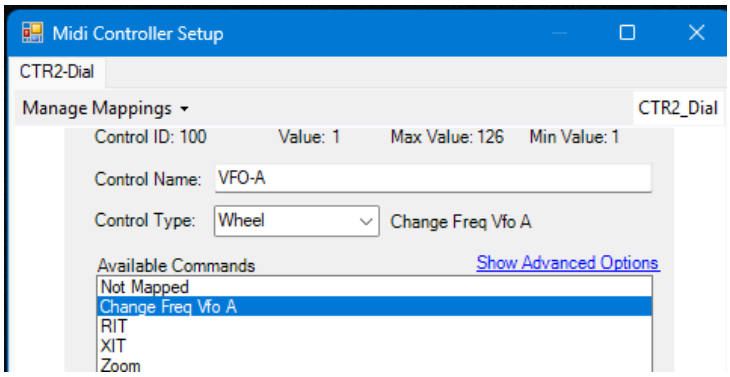
Mapping Controls

You can download a default Thetis map from the Lynovation web site. Once you have unzipped the Thetis file go to the **Midi Controller Setup** page in the app select **Manage Mappings -> Import Mappings** and select the Thetis map you unzipped.

To manually map your **CTR2-Dial** to Thetis, or change the default mapping, use the following process:

Touch a button or change a *dial* or *knob* setting and the window at the left will open.

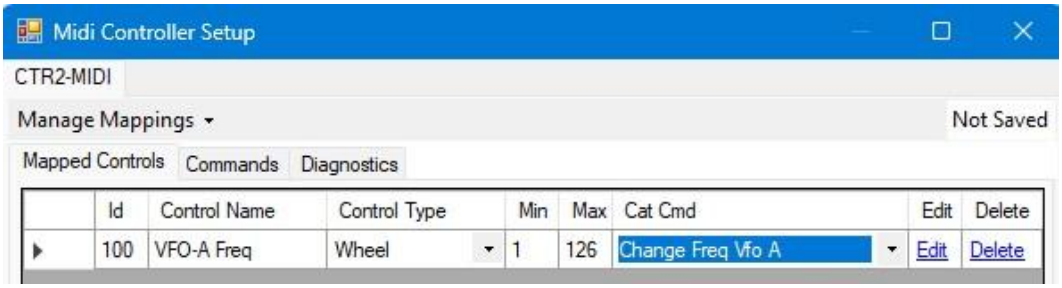
In this example I changed the dial control when it was set to the first control (VFO-A, MIDI control 100).



NOTE: When mapping *dial controls* you need to run the dial fully CCW until you get to 1 and then fully CW until you get to 127 to set the Min and Max range values.

Next, give the control a name. I'll call it **VFO-A Freq** and select **Change Freq VFO A** from the **Available Commands** list.

Click the **Done** and then the **Save** button to return to the **Manage Mappings** page.



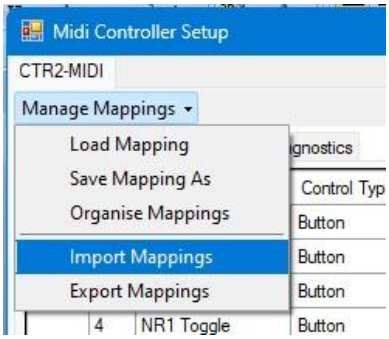
Now follow the same procedure to the other *dials*, *knobs*, and *buttons* in **CTR2-Dial**.

NOTE: When using the Thetis **Wheel** control type you must set the encoder on **CTR2-Dial** to **WheelB**.

When you're finished mapping controls make sure to click the **Save** button in the **Midi Controller Setup** page and **Apply** before you leave the **Setup** page so these functions will be restored the next time you run Thetis.

Managing Thetis Maps

You can save or load your map setting from the Thetis database using the **Load Mapping** and **Save Mapping As** options on the **Manage Mappings** tab. To load a map from others, use the **Import Mappings** option. To save your map to share with others, use the **Export Mappings** option.



NOTE: You need to open the **Configure MIDI** page in Thetis every time you start it in order for it to find **CTR2-Dial.**

Advanced Options

Operation with N1MM Logger+

You can use **CTR2-Dial** as an interface to use N1MM's CW keyer with SmartSDR or SDR-Control for iOS. To do this, follow the instructions on the N1MM Interfacing Basics webpage to build a key interface circuit. This will require a USB serial adapter with a keying interface circuit wired to its RTS output. Connect the output of the keying interface circuit to the TIP and SHIELD of **CTR2-Dial's** *Paddle input* jack. Next, map the *Left Paddle input* on **CTR2-Dial** to **Trigger CW Straight Key**. With this setup you can control your radio using the N1MM Logger+ and use keyboard keying in N1MM Logger+.

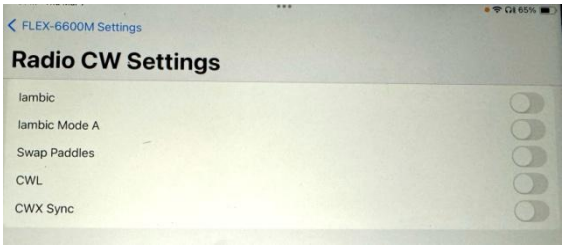
The biggest advantage of using a **CTR2** controller over using a mouse to control your radio is that commands from **CTR2** units go directly to your radio control app whereas your control app must be "in focus" to use a mouse. This is especially handy when running logging programs like N1MM Logger.

Unfortunately, this configuration doesn't allow you to use a set of paddles with the *MIDI* at the same time nor does it allow N1MM Logger+ to control your radio remotely.

If you're looking to have N1MM Logger control your radio remotely you might consider using **CTR2-Micro** to remotely control and key your radio with SmartSDR or SDR-Control providing the user interface. Instructions for this can be found in the CTR2-Micro Operation Manual under the **Advanced Options** section.

Troubleshooting

Issue	Solution
I can't connect the app to the **CTR2-Dial** using USB MIDI	1) Make sure you are using a USB-C data cable. Many USB-C cables are power only. These will not work with USB MIDI. 2) If using a mobile device (iPhone, iPad, Android, etc) you must use an OTG adapter for your device. This adapter provides a USB connection for an external device. Newer iPhones and iPads with USB-C connectors do not require an OTG adapter.
I can't connect the app to the **CTR2-Dial** Bluetooth LE device	Make sure the Bluetooth radio is turned on and the app has permission to use Bluetooth. Follow the instructions in the Connecting CTR2-Dial to the App and Enable Permissions sections. **NOTE:** Apple devices will not find **CTR2-Dial** in Bluetooth Settings
I have to connect **CTR2-Dial** every time I start the app.	**CTR2-Dial** will stay connected to the app on your iOS or Mac device with Bluetooth-LE as long as it remains powered up. If you power down the **CTR2** unit you'll need to open the **Tools -> CTR2 Dial Controller** window and click the **Find CTR2 Dial Bluetooth-LE Device** button then click **Enable** to reconnect it to the app. When connecting with USB MIDI it should say connected even through a power cycle. **NOTE:** You will need to open the **MIDI Configuration** window in Thetis every time you open it so it will find **CTR2-Dial**.
My Flex radio starts sending a string of Dits or Dahs when I select CW mode	This seems to be a bug SmartSDR v6.9.15. If you press the paddles when the app is not in CW mode but the CWX panel is open that paddle press is buffered in the app. When you change to CW mode from another mode, CWX triggers the last paddle press you made and keeps sending it until you press the paddle again, turn off CW mode, or close the CWX panel. This should be fixed in the latest update of SmartSDR.
CTR2-Dial connects to my device but does not control anything	1) Verify that you have mapped each MIDI control to a function in your app.
Frequency tuning is not working	1) Make sure you have the correct map selected in the **Settings** page. 2) Use **WheelA** for RHR and Marcus' iOS and MacOS apps. Use **WheelB** for Thetis and other Windows apps. 3) Make sure you have the correct MIDI command mapped to the frequency control in your app.

How can I tell when I'm at the center (0 Hz) when controlling RIT or XIT?	Use the **SliderB** control for **RIT** and **XIT** controls. The display will show a black bar indicating the offset and **CTR2-Dial** will beep and pauses tuning for about ½ second when the control reaches its center position (64 counts). **SliderA** does not beep at center or show the +/- offset.
Speed tuning is not working	Make sure **Speed Tune** is set to **On** in page 1 of the **Settings** page.
Paddles don't key the radio	1) Verify the radio is in CW mode 2) Verify the keyer is in **Breakin** mode and **Sidetone** in **On** in the P/CW panel 3) Verify **Pdl->** in the *Settings menu* is set the same MIDI control #s that you have mapped in the app for left and right paddle control. 4) Open the CWX Panel in the app's **View** menu 5) If you're using a Flex radio, it might be in CW Zombie mode.
Paddles are reversed	Remap the MIDI control #s to swap Left and Right paddle assignments
No sidetone in SmartSDR when keying	Open the left pop-out window in the Panadapter display on the app and click the **Audio** menu. Set the **Local Audio Monitor** slider to 0.
Slow response or timing issues with keyer	Connecting multiple Bluetooth devices to your iOS device (i.e. CTR2-Dial and a BT headset) may affect the app's keyer response to paddle input. To fix this problem, click the **Flex 6xxx** button on the bottom of the display, select the **CW** item to open the Radio CW Settings window and disable **CWX Sync**.  You may also consider using USB MIDI with an OTG adapter on your mobile device.
Frequency does not change correctly when I tune the *Dial* or *Knob*	First, make sure you have mapped the MIDI control # you are changing to your app's frequency control function. Second, there are two MIDI control types used to tune frequency, **WheelA** and **WheelB**. You must select the correct type for your app. • **WheelA** is used in Marcus' iOS/MacOS apps and in RHR. • **WheelB** is used in Thetis and other Windows programs If the wheel is the wrong type change it by connecting your **CTR2-Dial** to a terminal program or use the **Edit Dial Setting** option in the *Settings menu*.
Operating the paddles on a single knob unit displays **Knob B's** control and the keyer does not work.	Port B on the M5Dial (on the old **CTR2-Dial** hardware) supports either paddle input or the second external encoder (**Knob B**). The **# Knobs:** setting in the **Settings** menu determines which device is connected to **Port B**. **NOTE:** Starting with v 1.01.00 a new **None** option has been added to the **# Knobs:** settings. This allows you to turn off all external devices connected to **Port B**. Options are **None, One** (for a single encoder), **Two** (for two encoders), or **Knobs: Swap A/B**. The paddle interface works in all modes.
Turning **Knob B** causes the **Transmit** indication to show.	See the explanation in the previous topic. Change the **# Knobs:** setting in **Settings** from **One** to **Two**.

The knobs (encoders) don't work.	**# Knobs** in the **Setting** menu is set to **None**. Open the *Settings menu* and change the **# Knobs:** setting to match your unit's configuration. You can also reverse the location of the **A** and **B** encoders on dual-encoder units using **Swap A/B**.
Can I change *Knob A* and *Knob B* positions?	Yes, on the dual-encoder unit you can swap the **A** and **B** encoder positions by selecting **Knobs: Swap A/B** in the *Settings menu*.
I can't get the M5Dial into programming mode	There are two buttons that must be pressed in sequence to force the M5Dial into programming mode. Press and hold the **DOWNLOAD** button (labeled **BTN** on the back of the M5Stamp processor), then press and release the **Reset** button next to the processor, then release the **DOWNLOAD** button. The **DOWNLOAD** (**BTN**) is under the label on the M5Stamp processor and is similar to the **Reset** button. You should be able to press this button through the label with your fingertip. If you can't, the lever may have slipped of the button. Peel up the label and use an Xacto knife to flick the lever back onto the button. If it's damaged, use a no-conductive toothpick to press the button. See Appendix B for more details.
The tuning controls don't work	1) Make sure you have selected the number of knobs on your unit in the Settings menu. 2) If the tuning control has been locked it will be "grayed out". To unlock the control, long-press on the control's label.

Appendix A: Connecting CTR2-Dial to Your App

This section describes how to connect **CTR2-Dial** to various apps. Once you are connected to the app see the How CTR2-Dial Works with Your Control App section to understand how **CTR2-Dial** controls work with the app.

Remote Ham Radio

To connect **CTR2-Dial** to Remote Ham Radio simply plug your **CTR2** unit, running **CTR2-Dial**, into a USB port on your computer then select the **CTR2-MIDI** option in the RHR console's **Tool** menu (hopefully a **CTR2-Dial** option will be added in the future). The **CTR2-MIDI** option is pre-configured **CTR2-MIDI** so you'll need to change the configuration of CTR2-Dial to match RHR. You can see how the *MIDI*'s controls are mapped to RHR in the Remote Ham Radio Configuration section.

RemoteTx

RemoteTx is a browser based remote control system. Instead of connecting to the browser using MIDI commands, **CTR2-Dial** connects to **RemoteTx** as an HID (Human Interface Device). This eliminates most of the problems users experience with MIDI control in browser, especially when refreshing a page.

NOTE: RemoteTx is a subscription service and requires a Chrome, Edge, or Opera based browser.

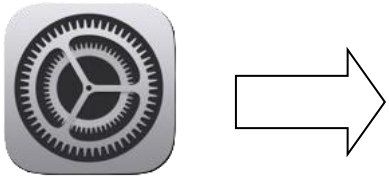
To connect the *MIDI* to RemoteTx, plug your **CTR2** unit running **CTR2-Dial** into a USB port on your computer and start the **RemoteTx** web page. Select the *Radio* button at the top of the page and click the **VFO** button. Next select the **M5STACK_DIAL** device from the dropdown list.

For information about setting your **CTR2** unit up to work with **RemoteTx**, see the RemoteTx Mapping section.

Marcus' Apps for iOS or MacOS

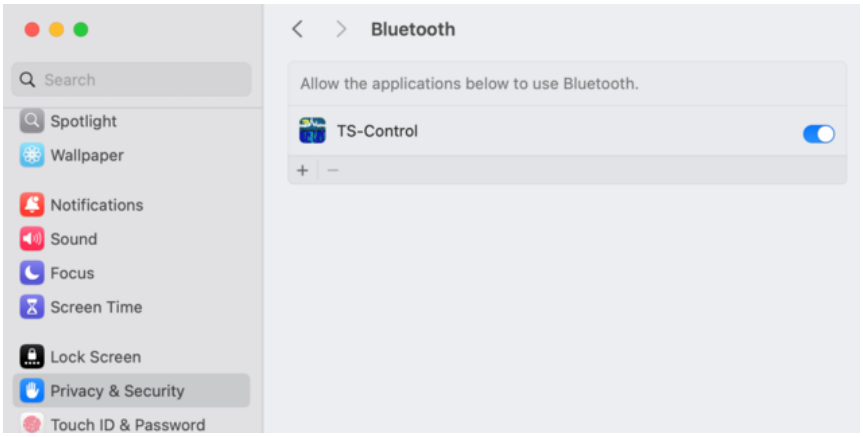
Enable Permissions

If you plan on using Bluetooth with Marcus' apps on your **iOS** device, open the app's **Settings** window and allow Marcus' apps access Bluetooth (not all devices have this menu).



You can also allow the app to use your iCloud Drive to save maps in this window.

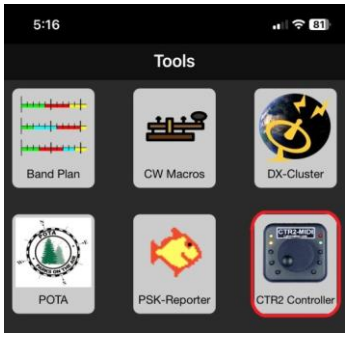
The app on **MacOS** will generally ask you if you want to enable Bluetooth the first time you start it. If you're already running SmartSDR, navigate to **Apple -> System Settings -> Privacy & Security -> Bluetooth.** This screenshot shows the *TS-Control* app has permission to use Bluetooth.

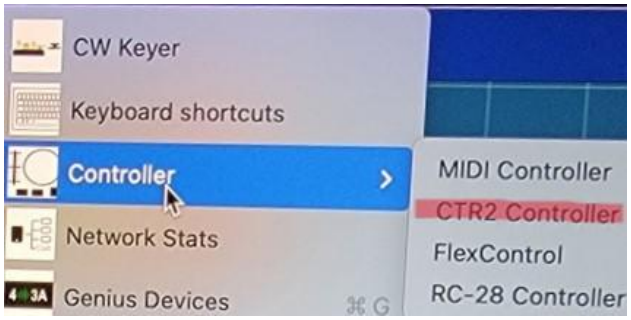


Select the CTR2 Controller Device

In Marcus' apps, select the **CTR2 Dial Controller** device in the **Tools** menu. The **Tools** menu will look different in each app so just look for the **CTR2 Dial Controller**.

Here's an iOS and MacOS example of the **Tools** menu. I've highlighted the **CTR2 Dial Controller** option in each menu…

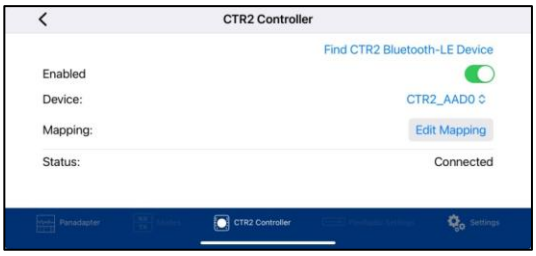




NOTE: These photos were taken before **CTR2 Dial Controller** was added to these apps.

CTR2 Dial Controller Screen

Once you select **CTR2 Dial Controller** from the **Tools** menu a screen like this will appear. It may look different depending on the Apple device you're using but it has basically the same information.



USB Wired Connection

To use your **CTR2** unit running **CTR2-Dial** with a wired USB connection on Mac's and newer iOS devices, connect your **CTR2** unit to your device using a USB data cable, select **STAMP_S3** from the **Device** menu, and click the **Enabled** button. The **Status** should change to **Connected**. Press the **Edit Mapping** button and press a button or turn the encoder to verify the app is receiving MIDI commands.

NOTE: To use a wired USB connection with an older iOS device (with a Lightning connector) you'll need to purchase a Lightning OTG (*On The Go*) adapter such as the one shown here. These are also referred to as "Camera Adapters" on Amazon. The adapter I use has two USB-A connectors on it. It also has a Lightning port so you can connect a wall charger or battery to power your phone and **CTR2** unit.

Bluetooth-LE Connection

To use Bluetooth-LE MIDI, make sure the Bluetooth radio is on (it is controlled in the *Settings menu*) then click the **Find CTR2 Bluetooth-LE Device** link to initiate a search for your **CTR2** unit running **CTR2-Dial**. The search will fail the first time you try to access a new device. If you get a failure notice, close the failure popup notification, **wait 10 seconds**, then click the **Find** link again. Once it registers your **CTR2** unit (your device's name is **CTR2_xxxx** with xxxx being the last four digits of the M5Dial's MAC address) click the **Enabled** button to start the Bluetooth connection. You don't need to select the name on the **Device** list unless you have two or more units running **CTR2-Dial** available.

NOTE: You can only have one **CTR2** unit connected via Bluetooth to the app at one time. You can use a second **CTR2** unit connected as a USB device. To do this, connect the Bluetooth **CTR2** unit using the **CTR2 Controller** device in the app's **Tools** menu and then connect the second **CTR2** unit using USB to the **CTR2 Dial Controller** device in the **Tools** menu.

If you can't get the app to connect to your **CTR2** unit, you probably haven't allowed the app to

use Bluetooth (see Enable Permissions above). If you still can't get it to connect, download **MIDI-Wrench** or **midimittr** for iOS or **Conji** for the Mac from the App store. They are free apps used to troubleshoot BLE MIDI issues. Connect your **CTR2-Dial** unit to one of these apps first and then try to connect again in the radio control app.

MacOS Bluetooth Connection Problems

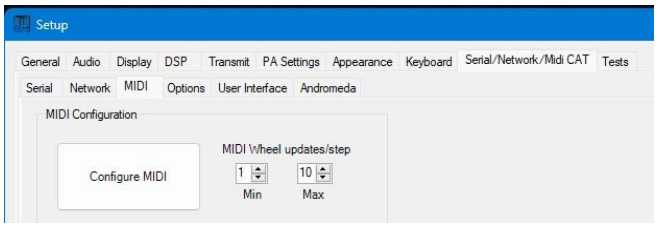
Follow these steps to connect **CTR2-Dial** to your Mac using Bluetooth:

1. Connect your **CTR2** unit running **CTR2-Dial** firmware to a USB port on your Mac.
2. In the **CTR Dial Controller** tool screen, check the **Enabled** button then click **Find CTR2 Bluetooth-LE Device** at the top of the screen.
3. The Mac may report that it can't find a Bluetooth-LE device. Wait 10 seconds the click **Find CTR2 Bluetooth-LE Device** again.
4. The Mac should report that it has found a Bluetooth device named **CTR2_xxxx** where xxxx is the address of your **CTR2** unit.
5. Select **Stamp_S3** on the **Device** list to use USB MIDI while connected to the Mac.
6. Select **CTR2_xxxx** on the **Device** list to use Bluetooth MIDI.
7. You may need to change the state of the **Enabled** checkbox to get your **CTR2** device to connect.
8. You can now unplug your **CTR2** unit from the Mac and use it remotely with Bluetooth.

Windows Applications

Thetis

Windows only supports USB MIDI connections. When you connect **CTR2-Dial** to your Windows computer's USB port it will automatically register as a MIDI device and be assigned a virtual COM port. Windows does not support Bluetooth-LE MIDI (at least not very well).



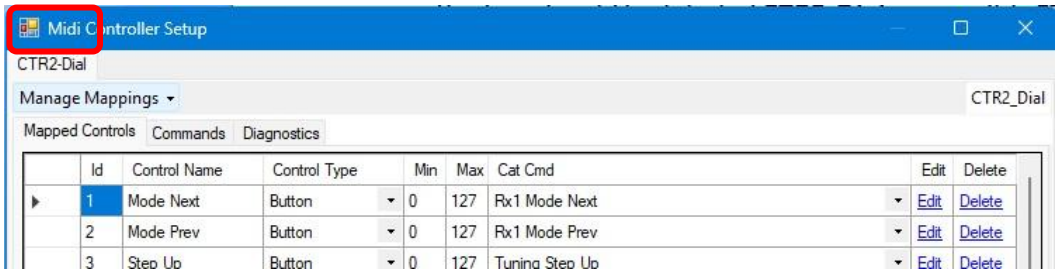
NOTE: USB MIDI requires a USB-C data cable. USB-C power cables, provided by many products, will not work!

Most Windows apps will need to be configured (i.e. mapped). **Thetis** is used as an example here but connecting to and mapping controls in other apps should be similar.

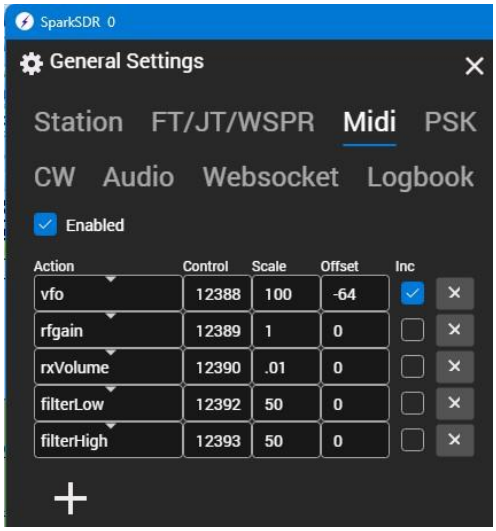
Additional information on mapping MIDI controls in Thetis can be found here.

Start by selecting the **Settings** menu on the main page in Thetis then select the **Serial/Network/MIDI CAT** tab. Click the **MIDI** tab as shown above.

Next, click the **Configure MIDI** button to open the **Midi Controller Setup** page. The program should go through an initialization sequence and end up on this page if it finds **CTR2-Dial** on the USB port. The tab at the top should be labeled **CTR2-Dial** or possibly **STAMP_S3**. If the initialization fails, double-check that you are using a USB-C data cable, not a charge cable and try again.



SparkSDR



SparkSDR (https://www.sparksdr.com/) has limited support for MIDI control. It doesn't support buttons, only the *dial* and *knob* controls. To map *CTR2-Dial* knob functions to this program, first plug your **CTR2** unit into a USB port on your computer and open SparkSDR. Once you have the program running, click on the **General Settings** menu and select **Midi**. You should see **CTR2-Dial 0** as a MIDI controller and turning the encoder should display MIDI messages from the controller. To map these messages to a control in the app simply press the **+** icon and enter the **Control** number shown in the **Last midi message:** window in the **Control** column. Then select an **Action** and set the **Scale** and **Offset** required for that control. You may need to play with these values to get the control action you want. Check the **Inc** box if you want the value from CTR2-Dial to be added to the current setting of the control. This is only used with the **vfo** control. This screenshot shows the settings I use.

Flex Radio CW Zombie Mode

If you plan on using your **CTR2** unit running **CTR2-Dial** firmware with a Flex radio using remote CW keying, there is an odd problem in Flex firmware where the radio can go into **CW zombie mode**. This happens occasionally when connecting and disconnecting multiple clients to the radio that send remote keying commands (at least I think that's what causes it).

In **zombie mode** the radio looks like it is transmitting CW, the light on the front panel turns red, the TX icon turns red, the MOX button turns blue *but **no RF carrier is generated** or displayed on the panadapter.* The only way I have found to get the radio out of zombie mode is to disconnect all clients and power cycle the radio.

Appendix B: Installing or Updating CTR2-Dial Firmware

CTR2-Dial is pre-installed as part of **CTR2-Flex** firmware on assembled **CTR2** units supplied with the M5Dial controller. If you're building your own hardware, or using your own M5Dial with a **CTR2-Quad Upgrade Kit**, you'll need to install the firmware yourself.

Starting with v2.00.00, **CTR2-Dial firmware became part of the CTR2-Flex firmware package**. As of v2.03.05, **CTR2-Flex/CTR2-Dial** firmware is now distributed as a single BIN file. This simplifies the installation process and reduces the possibilities of entering the wrong offset address for individual BIN files. The address of the single BIN file always starts at **0x0**. Instructions for switching between the two firmwares can be found here.

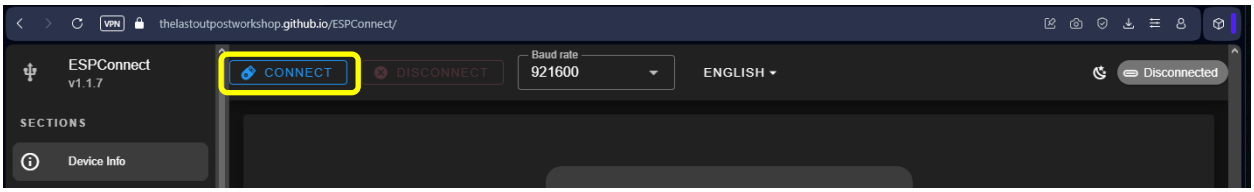
Before you begin, download the latest **CTR2-Flex/CTR2-Dial** firmware from my web site. Unzip the firmware distribution zip file into a folder.

Installing Firmware using ESPConnect (New Method)

By far, the easiest, and recommended method to install or update firmware on an ESP32 process is by using ESPConnect, an open source ESP32 management project.

ESPConnect is a browser-based tool that must be opened on a Chromium-based browser that supports Web Serial, such as Chrome, Edge, or Opera. It will run on a Windows PC or an Apple Mac. Unfortunately, Linux doesn't support Web Serial in any browser.

To start, open **ESPConnect** here: https://thelastoutpostworkshop.github.io/ESPConnect/



Next, plug your **CTR2** unit into a USB port on your computer.

NOTE: Do not connect it to an unpowered USB hub.

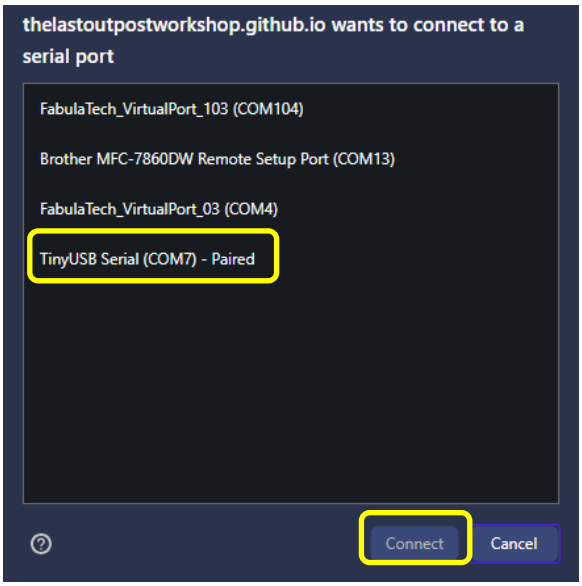
Click the **Connect** button.

A device list will pop up showing the available USB devices on your computer.

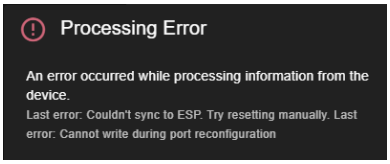
On a Windows PCs, all **CTR2** units will be shown as a **TinyUSB Serial** device, as shown here.

On Macs, **CTR2** units based on the M5Dial will be listed as an **M5STACK_DIAL** device.

Select the **CTR2** device and click the **Connect** button.



A popup warning will appear telling you that an error occurred. Some errors may tell you to press the BOOT button on the processor to put it into bootloader mode. <u>You do not need to do this.</u>



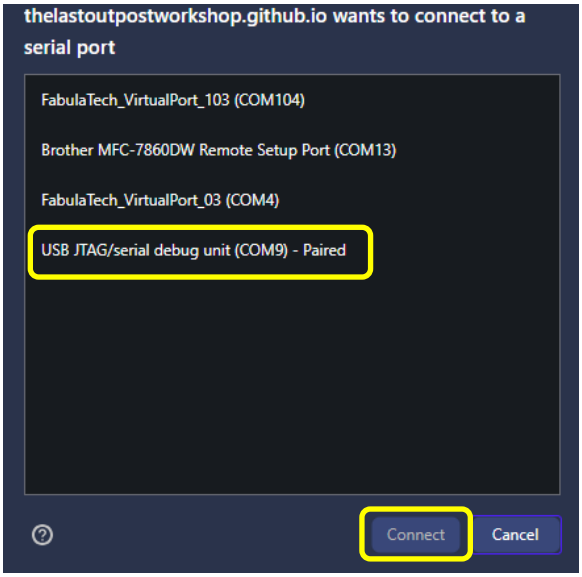
This is the normal response when trying to connect to a **Tiny USB serial** device.

Click the **Connect** button on the home page again.

This time, the **TinyUSB serial** (or **M5STACK_DIAL** on Macs) will not be shown on the list. Instead, a new device, **USB JTAG/serial debug unit** will be shown (on both PCs and the Mac).

This is the bootloader port and the unit is now ready to program.

Select the **USB JTAG/serial debug unit** device then click **Connect**.



Once you're connected to your **CTR2** unit, the left menu items on the page will be enabled.

<<< WARNING >>>

ESPConnect is a powerful device editor. Many of the functions can cause problems with **CTR2** firmware if you change them. If you don't know what a functions does, DON'T CHANGE IT!

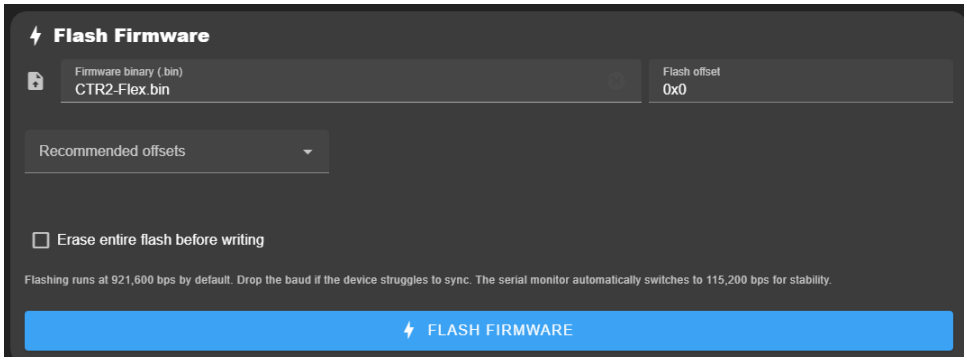
Click on the **Flash Tools** menu item and scroll down to the **Flash Firmware** section.

Click the **Firmware binary (.bin)** text box.

A file navigation screen will appear. Navigate to where you unzipped the firmware zip file from the firmware update page and click on the **CTR2-Flex.bin** file.

Your display should now look like this:

Leave the **Flash offset** set to **0x0** and uncheck the **Erase entire flash before writing** checkbox.



Click the **FLASH FIRMWARE** button to start the flash process.

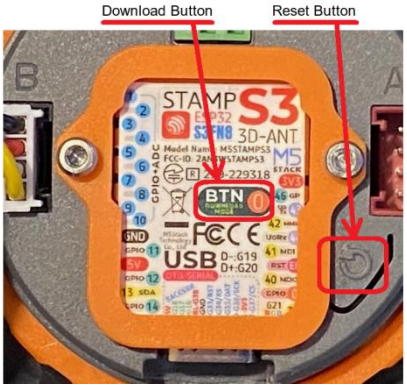
Once the process completes, cycle the power on your **CTR2** unit. The version # will be displayed when the unit boots. The version # should match the version # from the firmware zip file.

This completes the firmware update process.

Installing Firmware using EspressIF Flash Download Tool (Old Method)

The EspressIF Flash Download Tool is the original flash tool provided by the makers of ESP32 micro controllers. It's geared for developers and many find it difficult to use.

This section will describe the process of using this tool to update your M5Dial. I highly recommend using **ESPConnect**, described above, instead.



You must force the M5Dial into programming mode before you can flash firmware to it. To do this, follow this procedure

1) Open the enclosure and locate the back of the M5Dail
    a. On 3D printed enclosures remove the four screws holding the base to the shell with a 2mm Allen wrench
    b. On the New Age enclosure remove the four rubber feet on the bottom by pulling them off then remove the four Phillips screws under the feet
2) Press and hold the **DOWNLOAD (BTN)** button on the M5Dial then apply power to CTR2-Dial. The display will be blank when the unit boots into programming mode. If the unit is already powered up, press and hold the **DOWNLOAD** button then press and release the **RESET** button. You should hear a beep from your computer as it registers the new USB COM port and the display will be blank.

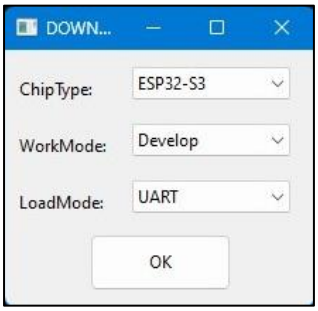
    NOTE: If you have problems getting the **DOWNLOAD** button to work its lever may have slipped off the button on the board. Remove the label and flick the lever back onto the button. Thanks Joe, KO8V for the tip!
3) You can now use the procedure below with EspressIF Flash Download tool (on Windows) or run the script (on Linux and Mac) with the new COM port to flash the new firmware to the M5Dial.
4) Once the download completes, remember to cycle the power on the unit to start the new firmware.

Now that you have the M5Dial in programming mode, follow these steps to install or update to the latest **CTR2-Flex/CTR2-Dial** firmware.

NOTE: EspressIF has released a browser-based flash tool called **ESPLaunchPad**. It can be used instead of using the Flash Downloader Tool described below. It works on Windows and Macs in several browsers. For more information on this tool visit https://ctr2.lynovation.com/espressif-browser-flash-download-tool/ Currently, Linux based browsers aren't supported in this tool.

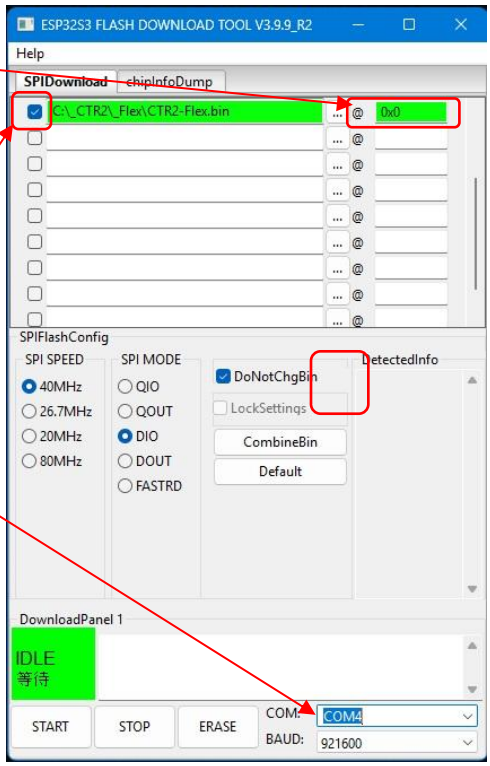
1. Download and unzip the latest CTR2-Dial firmware from **my web site.** Unzip that file into a different folder than where you store other Lynovation firmware update files.

2. Download and open the **EspressIF Flash Downloader Tool**. When it starts, select the **ESP32-S3 Chip Type**. Leave **WorkMode** set to *Develop* and **LoadMode** set to *UART.*



3. Map the **CTR2-Flex.bin** file that you unzipped from the **CTR2-Flex_v2.xx.xx** firmware distribution file into the downloader tool and set the offset address to **0x0**.
4. Select the checkbox on the left of the **CTR2-Flex.bin** filename as shown.

5. Set the **COM:** port to the port assigned to **CTR2-Flex/Dial** and set the **Baud:** to 921600.



NOTE: You must use a USB-C data cable. USB-C charge-only cables (supplied with may devices) will not work.

6. **On new installs**, click the **ERASE** button to clear all files from the M5Dial's memory.

NOTE 1: If you erase the flash memory after running CTR2-Dial firmware you will need to re-enter your call sign and registration key the next time you start CTR2-Dial.

NOTE 2: If you erase the memory after running the **CTR2-Dial** firmware you will need to re-enter your call sign and registration key and re-select the number of knobs on your unit the next time you start **CTR2-Dial**. If you previously exported your settings, you can restore them using the Import Settings option in the Settings menu.

7. Click the **START** button to download the firmware.

8. Once the download is complete, cycle the power on the unit to start the new CTR2-Dial firmware.
   NOTE: The ESP32-S3 processor will revert back to its operating COM port after the reboot.

Installing using a Mac or Linux Computer

Mac users should use the new method using ESPConnect, described above. Linux users have no other option, other than to follow the procedure below.

A script file is supplied in the firmware update zip file. This script file can be used in a Linux or Mac environment if you don't have access to a Windows computer.

Instructions for using this script file are included in the CTR2-Micro Operation Manual in **Appendix B**.

The firmware that allows USB MIDI control changes the way the virtual COM port works on the ESP32-S3. One COM port is used in the normal operating mode and another COM port is used for flashing the firmware. In order to flash new firmware to the unit you must force the ESP32-S3 to switch from the normal operating mode to programming mode.

To use the script file:

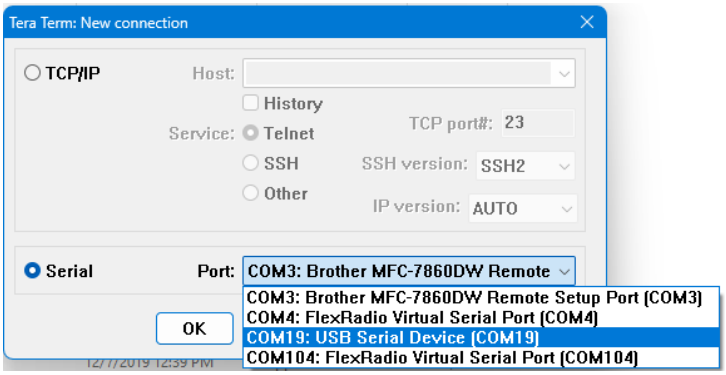
1) Force the M5Dial into programming mode as described above.
2) Edit the COM port in the script to the programming port. This port will be different than the operating COM port.
3) Edit the path to the .BIN file in the script.
4) Run the script.

Appendix C: Configuring Tera Term

Tera Term is the simplest terminal program to get running for a serial connection.

If you search for Tera Term you find a lot of garbage with malware attached to it. I've downloaded a clean copy of Tera Term v4.106 and posted it in the **CTR2 Group IO files section**. You can download it here. As far as I know, Tera Term is only available for Windows.

When you first open Tera Term you'll be presented with the **Tera Term New connection** window. Simply select the **Serial** radio button, select the COM port Window's assigned to your Micro when you plugged it in, and click the **OK** button.



Since you are connecting to a USB serial port there is no need to set the baud rate. It will run at USB speed regardless of the baud setting.

That's it! Tera Term will connect to **CTR2-Dial**. Press any key to start **CTR2-Dial's** terminal server.

You can change the terminal size in the **Setup** menu. Select **Terminal…** Set the **Terminal Size** to *41 x 20.* **CTR2-Dial's** terminal interface was designed for this size.

While in the **Terminal…** settings verify the **New-line** options are set to *CR* for both **Transmit** and **Receive** and the **Terminal ID** is set to *VT100*.

You'll probably want to change the font size and colors. These are also changed in Tera Term's **Setup** menu. Select **Display** to change the font and background colors to your liking. Select **Font** to change the font and font size. I like *Courier New*, *Regular*, and *14-point size*. Your preferences may differ.

Once you have the program configured the way you like, select the **Setup->Save Setup…** menu and save your configuration. If you use the default file name, TERATERM.INI the program will automatically start a Telnet session using the COM port you selected above when it opens. This provides one-click access to your **CTR2-Dial** firmware.

Appendix D: Configuring Putty

Putty is a terminal program that can be configured for a variety of needs. **CTR2** controllers based on the M5Dial only supports serial connections. This section describes how to configure the program to interface with **CTR2-Dial**.

NOTE: Putty does not have XModem transfer capability. If you want to export and import backup files for CTR2-Dial's settings consider using Tera Term or install ExtraPutty.

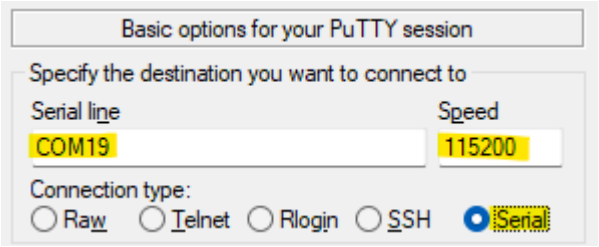
Download Putty for Windows from https://www.putty.org/. It's also available for Linux at https://www.ssh.com/academy/ssh/putty/linux and for Mac at https://www.ssh.com/academy/ssh/putty/mac.

You'll need to connect to CTR2-Dial using its USB serial port in order to configure it.

Serial Session

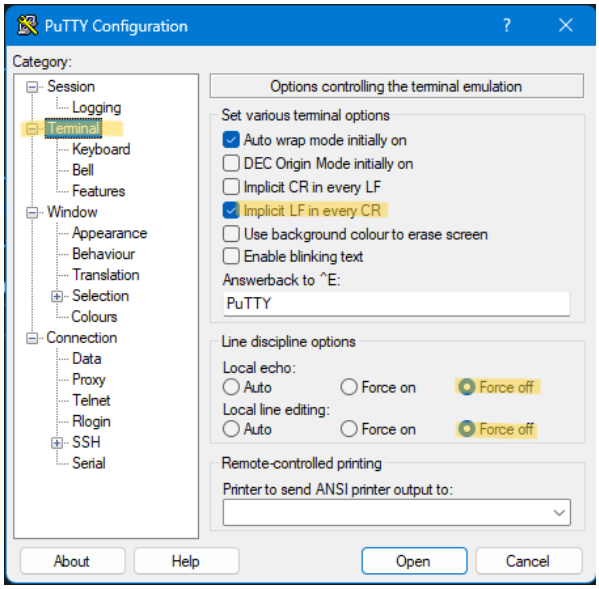
Select **Serial** then set the **Serial Line** to the COM port you found in the Device Manager and set **Speed** (Baud Rate) to 115200.

NOTE: Since this is a USB serial port the **Speed** (baud rate) doesn't matter. Data will be sent at USB speeds regardless of the **Speed** setting.

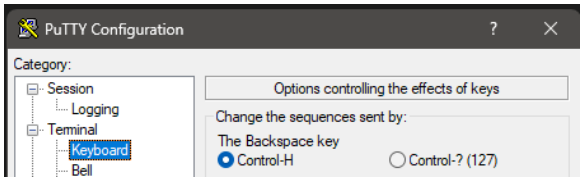


Next, select the **Terminal** item and set the **Implicit LF in Every CR** to on, and **Local Echo,** and **Local Line Editing** to **Force Off**.

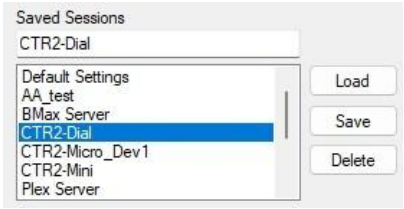
You can change the window size under the **Window** item. Set the **Columns** to *41* and the **Rows** to *20*.



Next, select the **Keyboard** menu option and note the setting for the **Backspace key.** If **Control-H** is selected you will need to press **Ctrl+Backspace** to send the **Del** key (ASCII 127). If **Control-? (127)** is selected, press the **Backspace** key to send the **Del** key code.



Once this has been done, return to the **Session** menu item, enter a name for this session and click the **Save** button. This allows you to easily re-open this session with just a couple of clicks.



If you right-click on the Putty icon in the Windows toolbar the last few sessions you had open will be displayed. Just select the one you want to open it.

You can adjust the display colors on the **Windows->Colours** menu item. The Micro uses the **Bold** attribute to highlight the ***hotkeys*** and other items. I like to set the **Background** color to blue and the **Bold** color to yellow but you can find the colors that work for you. After you get a color combination you like return to the **Session** menu and **Save** the session.

Appendix E: Apple or Linux Terminal Programs

The Apple Mac and Linux have built-in terminal programs so there is no need to install a separate app. To connect your **CTR2** unit to a terminal session, use the following process.

First, list your current serial ports without your **CTR2** unit plugged in.

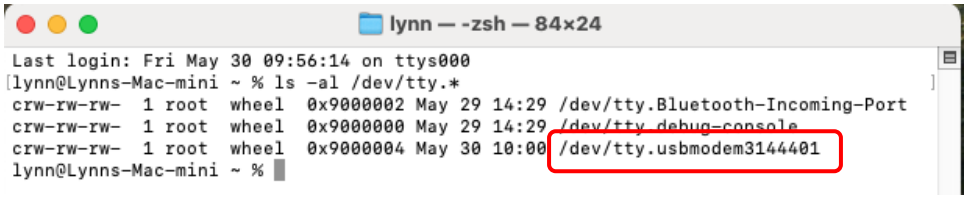
- On the Mac open **Applications/Utilities/Terminal.app**. On Linux open the terminal program supplied by your distro.
- On the Mac, enter ***ls -l /dev/tty.usb****, on Linux, enter ***ls /dev/tty**** This will return a list of all known serial ports.
- Next, plug the *Flex* into the computer's serial port and execute the command above again. This is easily done by pressing the *Up* arrow key.
- Compare the new list with the old list. The *Flex's* serial port ID will appear on just the new list. For Mac users the serial port ID format will be **/dev/tty.usbserialxxxxx** where **xxxxx** is a unique device ID #. Linux users will see something like **/dev/ttyACMx** or **/dev/ttyUSBx**, where **x** is a unique # for that port.

NOTE: If a new virtual serial port is not created when you plug your Micro into your PC make sure you are using a USB-C cable that supports data. Many USB-C cables only provide power to the remote device.

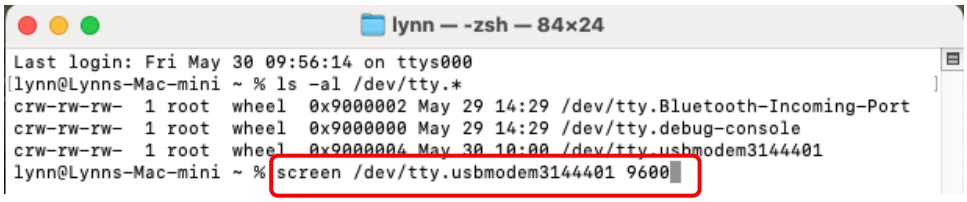
Once you know your **CTR2** unit's USB serial port ID, write it on the label on the bottom of the unit using a fine-tipped permanent marker for future reference. Put a piece of transparent tape over the label to seal the ink so it doesn't rub off (it's not as permanent as you think). You can always remove the tape if you want to change what's written on the label.

Once you have the serial port ID, enter the following: **screen {serial port ID} 9600**. Include the complete device description (i.e. **/dev/ttyxxxxxxx**) for the **serial port ID**. This will open the serial port using 9600 baud in a terminal session. The following screenshots demonstrate these steps.

1. Get the list of serial devices on your computer. We're looking for the **usb** device.



2. Open the **screen** utility using the **usbmodem31444-1** device. Your USB device will have a different numerical #.



3. This is CTR2-Dial's terminal display. Select a dial control or button to edit that control's label.

Appendix F – Power Requirements

The approximate current on an M5Dial equipped **CTR2** unit's USB connection is shown below for various configurations. If you've installed the 3.7-volt lithium battery option, these numbers will help you determine the length of time your unit will run on the battery. The M5Dial doesn't have the hardware necessary to monitor the battery's state of change, so you won't know when the battery is about to be depleted.

Bluetooth Mode	Backlight	Current
On	High	170 mA
	Medium 2	152 mA
	Medium 1	133 mA
	Low	115 mA
Off	High	121 mA
	Medium 2	105 mA
	Medium 1	82 mA
	Low	70 mA

By comparison, because it has no display, **CTR2-MIDI** draws 93 mA with the Bluetooth radio on and 43 mA with it off.

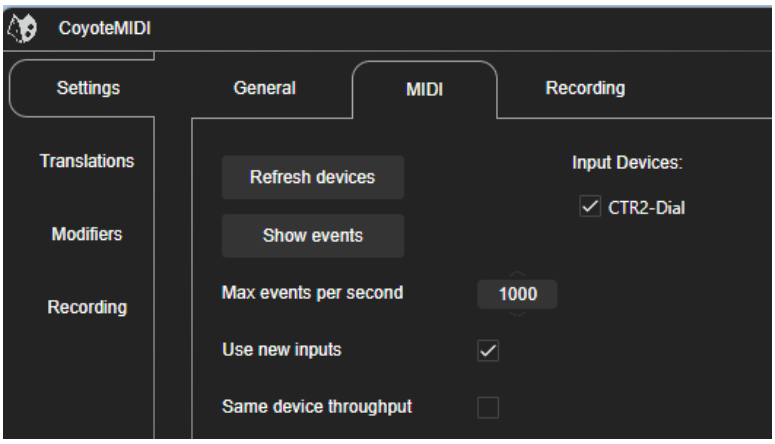
Appendix G: Using CTR2-Dial with Non-MIDI Programs

CTR2-Dial firmware is not limited to programs that support MIDI controls. Third-party MIDI translator apps such as CoyoteMIDI can receive MIDI commands from **CTR2-Dial** and convert them to other actions such as mouse scroll or keystroke actions. MIDI translator programs are available for Windows, Mac, and Linux.

These apps allow **CTR2-Dial** to be used as a physical tuning knob with radio control programs such as SmartSDR for Windows, wfView, and WebSDR that don't normally support external tuning knobs. And it doesn't stop there. These apps also allow **CTR2-Dial** to be used with any program that accepts mouse scrolls or keystroke for control. For instance, it makes a great jog wheel for Audacity.

There are a couple of drawbacks to using a translation app instead of native MIDI control. First, like your mouse or keyboard, the app you want to control must be "in focus" to receive the mouse scrolls or keystrokes from the translation app. Second, most programs don't have a lot of control functions tied to mouse scroll or keystroke actions. For instance, SmartSDR for Windows only accepts mouse scroll actions for frequency control and wfView only supports the J and K keys for frequency control.

Configuring CoyoteMIDI



This section will cover configuring CoyoteMIDI to interface with SmartSDR for Windows and wfView. The translator you choose will have a different setup procedure but they're all basically the same.

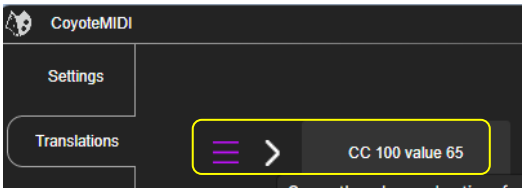
Before you start, plug your **CTR2** unit running **CTR2-Dial** firmware into a USB port on your computer. It will automatically register as a MIDI device. Open the **Dial** control and turn the gray ring to select the first control, typically named **VFO**. There

Once you've installed CoyoteMIDI open the app and navigate to the **Settings -> MIDI** tab. **CTR2-Dial** should show up on the **Input Devices:** list. Click the checkbox to enable it.

Configure SmartSDR for Windows Frequency Control

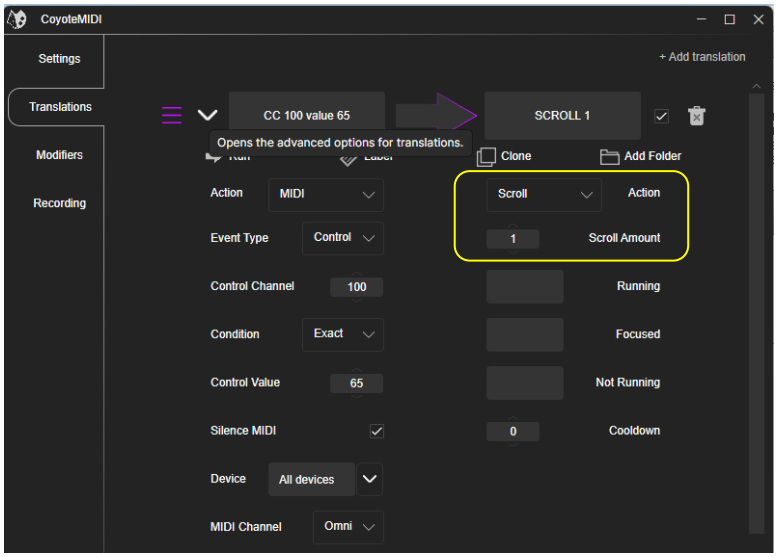
Next, select the **Translations** tab and click the **+ Add Translation** button in the top-right corner.

The first example uses the **VFO** dial control (MIDI control #100). This mode uses the **WheelA** MIDI control where a value of 65 represents a clockwise turn and 63 represents a counter-clockwise turn of the knob.



Click the box with **Click to record** then turn the knob on CTR2-Dial *slowly* clockwise. You should see **CC 100 value 65** appear in the left box. If you don't see this, make sure the selected control on CTR2-Dial shows control 100 above the control's label. Hover the mouse over this entry to show the violet menu and **>** icons.

Click the **>** icon to view the details of this command. I've already set this command to be translated to the **SCROLL 1** mouse command.



The options on the left should be as shown. If not, edit them to match what's shown here.

To configure the **SCROLL** 1 output, click the **Action** dropdown box and select **Scroll**. Next, set the **Scroll Amount** to **1** so each tick of the clockwise turn of the encoder produces on scroll tick.

That's it. You configured your first translation. Click the **+Add Translation** button again and follow the same process to program the counter-clockwise knob turn. In this case, set the **Scroll Amount** to **-1**. This will send the reverse mouse scroll.

Now open SmartSDR for Windows and the frequency should change as you change the knob on CTR2-Dial. You can edit the label on CTR2-Dial with a terminal program if you want to change it.

Configure wfView Frequency Control

You can set up translations for many other programs the same way. Next, we'll go over setting up keystroke "J" and "K" to control the frequency in wfView.

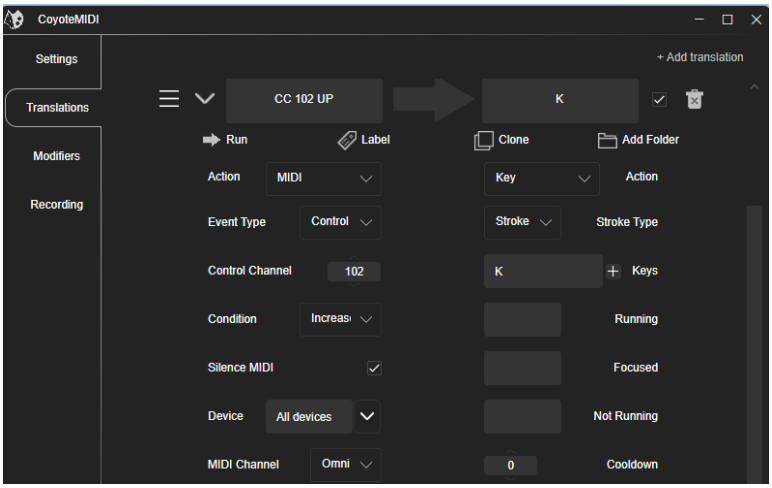
Turn the gray ring on CTR2-Dial to switch to control 102. This is normally shown as **AGC-T**. You can change this label with a terminal program. You can also use any other knob control.

NOTE: This encoder uses the **SliderA** control type. This type operates like a potentiometer and has a range of 0 to 127 counts. To use this control type we'll look for the value increasing or decreasing to determine the direction the knob is turning. You can edit the MIDI control type in a terminal if you want to change it to **WheelA**. If you do this, follow the setting in the first example.

Click the **+Add Translation** button to add another translation.

Click the box that says **Click to record** then turn the knob on the *MIDI* clockwise. You should see **CC 102 value xx**, where xx is a number between 1 and 127.

Hover the mouse next to this box then click the **>** icon to show the details.



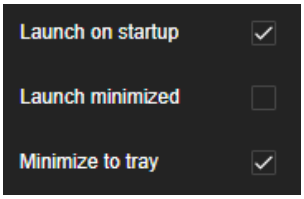
On the left column, click the **Condition** dropdown box and select **Increase**. This action will trigger when the MIDI value is increasing. The title in the top box will change to **CC 102 UP**.

In the **Action** dropdown box in the right column select **Key** and in the **Stroke Type** select **Stroke**. Finally, in the **+ Keys** box, enter **K**. This is the key used to increment the frequency in wfView.

Next, click the **+ Add Translation** button again, click the new box showing **Click to record** and turn CTR2-Dial's knob counter-clockwise. Now, follow the same process as outlined above except select **Decrease** in the **Condition** box and enter the letter **J** in the **+ Keys** box. wfView uses the **J** key to decrement the frequency. I'm not aware of any other keystrokes wfView supports. If there are more they can easily be added.

CoyoteMIDI Startup Options

There are a couple of options in the **Settings -> General** menu that you might want to set if you regularly use the *MIDI* and CoyoteMIDI with your apps. If you select **Launch on startup** and **Minimize to tray** options CoyoteMIDI will automatically start when you boot your computer and will be available in the Windows tray if you need to change anything.



Appendix H: Change Log

v2.03.06: December 26, 2025

- Additional code refinements to paddle debouncing

v2.03.05: November 10, 2025

- Firmware is now released as a single .BIN file instead of four .BIN files
- Added information on switching between CTR2-Flex and CTR2-Dial firmware modes
- Added paddle release debounce to fix issues with cheaper paddles adding extra code elements
- Improved keyer Bug mode so it works properly now
- Fixed several bugs in the Export and Import Settings functions
    - WiFi credentials and radio IP settings are now encrypted in exported settings files

v2.03.04: October 9, 2025

- Fixed bug that caused Dial/Flex modes to switch when running unit on lithium batteries

v2.03.03: September 16, 2025

- Added Touch Delay On/Off to **Setting** menu to enable or disable the 120-millisecond delay on touch events that was added in v2.03.02
- Added Paddle Normal/Reverse to Setting menu to roll paddle wiring**v2.03.02: September 10, 2025**
- Added 120 mSec debounce to screen touch – Some M5Dials send random touch events
- NOTE: v2.03.01 was released for one day – removed it to fixe bugs in debounce code

v2.03.00: September 4, 2025

- You can now enter your registration credentials without using a terminal program.

v2.02.00: August 18, 2025

- You can now edit control labels (and MIDI types for **Dial** controls) without using a terminal program.
- Added **Ring Control** options – you can now assign a dial function to the gray ring encoder on the display
- Added a Tuning Lock option

v2.01.01: August 10, 2025

- You can now choose whether or not you want to reset your call and registration key in the Reset to Factory option.
- Updated the Dial and Button labels in Map 1 so they match the MIDI function mapping in SmartSDR for iOS/MacOS.
  NOTE: You will need to reset your unit to Factory settings to update the labels on you unit.

v2.01.00 – August 9, 2025

- Removed the **Maps and More Maps** section and replaced it with **How CTR2-Dial Works with Your Control App**. Also rearranged and edited sections to make it easier to use this manual.

v2.01.00 – August 7, 2025

- Added the ability to assign encoder switches on CTR2-Flex hardware to virtual buttons
- Added additional information about mapping CTR2-Dial to your control app
- Added a new link to download Tera Term. Get it from the CTR2 Group IO File folder here.

v2.00.00 – July 24, 2025

• Integrated CTR2-Dial firmware into CTR2-Flex v2.00.00 firmware

- This gives you the ability to run either CTR2-Dial MIDI firmware or CTR2-Flex WiFi firmware on the same device
- CTR2-Flex v2 runs on CTR2-Dial and CTR2-Flex hardware
- No functional changes were made to CTR2-Dial firmware when it was integrated into CTR2-Flex v2 with the exception of a few Settings menu items being moved around
- Your current CTR2-Dial v1 settings will automatically be converted and saved to CTR2-Flex v2
- You can run CTR2-Flex from the Settings menu
- If you don't run a Flex 6000 or 8000 radio and/or don't have CTR2-Flex hardware, you can continue running CTR2-Dial v1.01.01

v1.01.01d – July 21, 2025

- Added additional information to Appendix B on preparing CTR2-Dial to flash firmware

v1.01.01c – June 29, 2025

- Added information about using EspressIF's new browser-based flash download tool ESPLaunchPad

v1.01.01b – May 30, 2025

- Added Appendix E with information on using the **screen** command in Mac and Linux for a terminal interface

v1.01.01a – May 22, 2025

- Added a note about using a USB-C data cable

v1.01.01 – May 14, 2025

- Fixed a bug in the new encoder swap code introduced in v1.01.00. Swapping encoders works now.
- Renamed the **Port B:** option in **Settings** to **# Knobs:** and changed the options from **Off, Paddles**, **Enc B** and **Swap A/B** to **None, One**, **Two**, and **Swap A/B**.
- Paddle input now works when **# Knobs** is set to **None**
- Added a note about problems with the **DOWNLOAD** button when flashing firmware
- Added Appendix G: Using CTR2-DialI with Non-MIDI Programs with examples of how to use CTR2-MIDI with programs like SmartSDR for Windows that do not support MIDI control

v1.01.00 – May 7, 2025

- Released for production

v1.01.00k – May 5, 2025

- Added the ability to edit the functions in the **Knob** menus without having a terminal connected
- Added [Knobs: Swap A/B] option for dual-encoder units
- Added Backspace key to start registration entry and added note about Del key mapping in Putty
- Open **Port B:/# Knobs** option in settings after successful registration key entry

v1.01.00f – May 1, 2025

- Added left/right swipe to the **Dial control** and the **Dial, Knob**, and **Buttons** menus to switch between pages.

v1.01.00e – April 30, 2025

- Increased encoder functions from 12 to 18 for each map
- Hide unlabeled encoder functions on Dial control unless user has connected to the unit with a terminal to edit the function labels.
- Updated the Dial and Knob graphics to minimize the flashing that was present in earlier versions.
- Added a **Port B: Off** option to Port B settings. This allows you to completely turn off Port B when not using the paddle or encoder B options. See this note about setting Port B on a new unit.
- Removed touch & hold speed acceleration on wheel controls – replaced with speed sensitive tuning on all controls.
- Added a new **Info** option in the Settings menu. This option displays the firmware version #, registered call sign, and the registration key for the firmware.
- Added support for the new port expander interface board. This board will support two external encoders and the paddle input jack.

v1.00.06 – January 30, 2025

- Fixed a major bug in the initialization file. I was not updating button labels above #16 so any changes made to buttons above this number would revert back to default on a power cycle. All users should update to this version.
- Fixed the code that allows you to swipe left or right on the **Buttons** menu to change pages

v1.00.05a – January 29, 2025

- Updated manual to include more information about connecting to Mac computers using Bluetooth
- Added troubleshooting information for issues cause by an incorrect setting of **Port B** in the **Settings** menu.

v1.00.05 – January 28, 2025

- Added “double-touch” to many of the caption and footer buttons. Touch once to select and again to execute
- Added left and right functions to some of the caption and footer buttons.

v1.00.04 – January 24, 2025

- Fixed bug causing incorrect MIDI control # to be displayed on Knob changes

v1.00.03 – January 23, 2025

- Fixed bug causing a single reverse pulse when tuning through 0 degrees on the Dial.

v1.00.02 – January 22, 2025

- Fixed minor bug in PTT page – touching **PTT** caption did not return you to the ***Home*** page
- Added path to Thetis MIDI maps to Thetis section

v1.00.01 – January 20, 2025 – Shipped version

- Added **WheelB-r** (reverse) wheel option to switch direction of dial rotation for PI HPSDR
- Touching the **CTR2-Dial** caption on the ***Home*** page returns to the previously open page
- Fixed bugs in **Knob B** property editing
- Added Power Requirements to Appendix F
- Changed the procedure to flash firmware to the M5Dial to require manually forcing the M5Dial into programming mode
- Added a new MIDI Control Types table

v1.00.00 – January 16, 2025 – Initial Release

- Updated photos with new enclosure
- Added Import/Export Settings options
- Allows single encoder to use all 12 Knob settings