KDØRC

**TeensyMaestro**

User Manual

Len Koppl, KDØRC

12/17/2020



1. Introduction

The TeensyMaestro is a Teensy (Arduino work-alike) based set of controls for the FlexRadio 6000 series. It provides knobs, buttons, display and a keyer. It is designed to augment SmartSDR (SSDR), not replace it. There are many functions that the TeensyMaestro cannot perform that still need to be done using SSDR or a real Maestro.

The TeensyMaestro uses a USB cable for power and an ethernet cable to connect with the radio. All communication with the radio is via TCP/IP, not the USB cable.

The primary design goal was to provide the most used controls in the form of knobs and buttons while retaining all the power and convenience of the computer-based SSDR. It uses the extensive TCP/IP Application Programming Interface (API) provided by FlexRadio. It also leverages the Flex Arduino library built by Enzo, IW7DMH. Without this library, the TeensyMaestro would not have been built.

While the TeensyMaestro has a color touch screen, it does not display a panadapter or waterfall. It does show basic slice information in a similar way as the “flag” does in SSDR. The touch screen is a rudimentary resistive type, so it does not have cell phone like capabilities but provides some simple functions. There are no functions that rely solely on the touch screen; the physical controls cover all touch screen functions.

1. Setup
   1. Connections and power

Power up the Flex and start SmartSDR. Plug the ethernet cable into the same network (LAN) as the Flex. If this is a DHCP-served router or modem (most are), the TeensyMaestro will obtain an IP address for connection to the radio. If you use a network switch to connect the computer (running SSDR), the radio and the TeensyMaestro together without benefit of DHCP (i.e. no modem/router), then you may or may not need to change settings on the SD card (see section 6.3, below).

Turn on the TeensyMaestro by plugging one end of a USB cable into the back of the TeensyMaestro and the other end into any USB port that provides 5 VDC. The TeensyMaestro does work without SmartSDR running and connected to the same Flex radio, but generally, that is not the preferred way to operate (see section 3.1 for instructions on operating standalone). The rest of setup will assume that the Flex is powered up (green light on steadily), SmartSDR is running and connected and that the TeensyMaestro is connected to the same network as the radio. It is also assumed that the TeensyMaestro is powered up *after* the radio has a solid green light and SmartSDR is running and connected to the radio.

* 1. Loading the Software

Before using the TeensyMaestro for the first time, the software must be loaded. Teensy.exe is used to load the .hex file (executable object code). Start Teensy.exe and plug the Teensy into an available USB port:



Click *File*, *Open HEX File* and select **MiniMaestroTeensy.ino.TEENSY41.hex** from whatever location you have it saved. Press the Program Load button on top of the Teensy board and the software will load and start.



USB

Use this SD Card slot

Program Load button

* 1. SD Card

The Teensy board has a micro-SD card slot that can be used to hold configuration information. Edit MMConfig.ini to fit your needs and copy it to a FAT32 formatted micro-SD card (see section 6, below for information on editing the file). A very small card can be used, as MMConfig.ini is the only file needed.

The display has an SD card slot on it, but it is not used in this application. Be sure to put the SD Card in the slot on the Teensy board itself, not the one on the display.

1. Controls
   1. Radio Controls

The front panel is generally laid out with Slice A controls to the left and Slice B controls to the right. All of the encoders have push buttons built in except for the VFO controls. The section below describes the function of the rotary encoder, followed by the short press action in single angle brackets (< >) followed by the long press action in double angle brackets (<< >>). In some cases, the encoder will take on a different function as indicated by the angle brackets.

* + 1. VOL A encoder <Mute> <<Select Slice>>

Adjust Slice A volume (0 – 100).

Short press mutes or unmutes Slice A.

Long press sets Slice A as the active slice. If Slice A is not in use, it creates Slice A and sets it to be the active slice.

* + 1. AGC-T A encoder <RIT Set> << XIT Set >>

Adjust Slice A AGC threshold (0 – 100). <Adjust Slice A RIT (-99999 – 99999)>. <<Adjust Slice A XIT (-99999 – 99999)>>.

Short press highlights the Slice A RIT and turns it on. Rotary encoder now adjusts RIT. After 5 seconds of no encoder operation or an additional long or short press, the highlight goes away and the encoder returns to its primary function.

Long press highlights the Slice A XIT and turns it on. Rotary encoder now adjusts XIT. After 5 seconds of no encoder operation or an additional long or short press, the highlight goes away and the encoder returns to its primary function.

* + 1. Low/Shift A encoder <NB> << NB Set >>

If mode is CW, shifts Slice A filter passband up or down in 10 Hz increments. If mode is not CW, adjusts filter low cut in 10 Hz increments. <<Adjust Slice A NB (0 – 100)>>.

Short press toggles Slice A NB on and off.

Long press highlights the Slice A NB and turns it on. Rotary encoder now adjusts the NB. After 5 seconds of no encoder operation or an additional long press, the highlight goes away and the encoder returns to its primary function.

* + 1. Hi/Width A encoder <NR> << NR Set >>

If mode is CW, adjusts Slice A filter passband width up or down in 10 Hz increments. If mode is not CW, adjusts filter high cut in 10 Hz increments. <<Adjust Slice A NR (0 – 100)>>.

Short press toggles Slice A NR on and off.

Long press highlights the Slice A NR and turns it on. Rotary encoder now adjusts the NR. After 5 seconds of no encoder operation or an additional long press, the highlight goes away and the encoder returns to its primary function.

* + 1. CW Speed encoder <Menu Select> << Menu Exit >>

Adjusts CW speed (5 – 100 WPM).

If not already in menu system:

Short or long press enters menu.

If in menu system:

Short press executes selected item. If no item is selected, short press exits menu system.

Long press exits menu system without executing any items, selected or not.

Five second press from any screen:

Will restart the Teensy MCU. If the TeensyMaestro is powered up before the radio, you can use this feature to restart instead of cycling power. Press and hold the Menu Select button until a blue screen appears with the message “Release to RESTART”.

* + 1. RIT A <Step Dn> <<Menu Dn>>.

Short press to toggle RIT A on or off. Long press to change VFO A step size down. Note that the TeensyMaestro VFO steps are independent from SmartSDR VFO steps.

If in the menu system, short press to navigate one menu item down.

* + 1. XIT A <Step Up>.

Short press to toggle XIT A on or off. Long press to change VFO A step size up.

* + 1. Slice B controls work in identical fashion as the A controls and will not be described separately here.
  1. Keyer Controls
     1. Six CW message buttons.

Short press activates messages 1 – 6. Long press activates messages 7 – 12. See CW Menu for more information on configuring the buttons to activate TeensyMaestro messages vs. Flex CWX messages.

* + 1. SmartSDR.



Turn off Iambic when using the TeensyMaestro keyer if it is plugged into the keyer jack or you will get all dots. If you connect the keyer to pin 4 of the 15 pin accessory connector, Iambic can be on or off to control the behavior of whatever you have plugged into the key jack.

1. Touch Display
   1. Display

The TeensyMaestro has a 3.5 inch 320 x 480 color touch screen. It shows similar information as the “flag” does in SmartSDR. It does not show a panadapter or waterfall and is designed to work *with* SSDR, not instead of it.



These green boxes indicate the approximate location of the touch zones and are not displayed on the screen.

* 1. Touch Screen

There are 5 touch zones on the display. Touching the slice name (A or B) will set that slice as the active slice and will gray out the inactive slice. If no slice is in use, touching the blank spot where the slice name would normally be will create that slice and set it to active. This function is not available on Flex versions less than 3.xx.

Touching a grayed-out TX indicator will set TX on for that slice and will gray out the other slice TX indicator. Long-touching a red TX indicator will turn it off and will not affect the other TX indicator.

Touching the screen in the large center area displays the splash screen as long as it is being touched. Information such as Teensy MAC address, IP address and Flex radio information like model and version is displayed.

The bottom information area of the screen does not respond to touch.

Note that touching the screen when the screen saver (blank screen) is active will restore the normal view and reset the screen save timer. Note also that this is a resistive, not a capacitive touch screen. This means that it can respond to simple touches, but is not capable of the intricate gestures that a cell phone or tablet screen are capable of. All touch functions are available using the physical controls, so it is not necessary to use the touch screen.

* 1. Display elements



The left side shows Slice A, the right shows Slice B. The bottom shows items not tied to a slice. If a slice is not in use, its side of the screen will be blank.

A yellow slice name (Slice B in this example) shows the active slice. Each slice can be operated with its controls regardless of active status. Active status is more important in SmartSDR for mouse wheel control of frequency.

The slice with TX highlighted in red indicates the transmit frequency.

NB, NR, RIT and XIT are active when yellow and inactive when gray.

Step shows VFO step size in Hz. Default is 10 for CW, 100 for everything else. See Configuration File below for more info on changing the default.

The filter graphics for CW show offset (0 Hz in this example) width (400 Hz) lower and upper bounds (-200 Hz and 200 Hz respectively). Note that any filter bound that is lower in frequency than the carrier shows as a negative number. Lower sideband (LSB) will show negative bounds and upper sideband (USB) will show positive. If a control takes a bound past the carrier frequency, it will change sign.

Across the bottom panel are:

* CW speed and active profile. If no profile is active, this will be blank.
* License class, CW Message Source and CW contest serial number.
* RF Power (only updates after a transmission), CW Paddle “handed-ness” and keyer mode.

1. Menu
   1. Navigation.

Press (Short or Long) the Menu Select button to enter the menu system. Turn the CW Speed encoder to scroll through menu items on a menu page. Selected items will be highlighted with a yellow box. Short press the Menu Select button to activate the selected item. Long press the Menu Select button to exit the menu without taking action. Depending on the item, activating it will leave the menu system, or leave you in the menu system to cycle through choices. If no items are selected (i.e. nothing is highlighted) a short or long press of the Menu Select button will exit the menu system without activating any items.

Use the Menu Up and Menu Dn buttons to scroll among menu pages.

* 1. Filter Menu

List of filters based on mode of the active slice.

* 1. Profile Menu

Lists all profiles in the radio. Multiple menu pages will be used if necessary.

* 1. Misc Menu
     1. Reload Config File

Reads the SD card, reapplies all values except SPI Clock so that a power cycle is not required if the card is edited. The SPI Clock value is not applied until power is cycled.

* + 1. Out of Band Display

ON or OFF. Default is ON, can be changed in config file.

* + 1. Restart TeensyMaestro

Equivalent to cycling power. Can also be restarted by holding the Menu Select button down for 2 seconds.

* 1. Memory Menu

List of memories in the radio. Multiple menu pages will be used if necessary.

* 1. CW Menu
     1. CW Message Source

TEENSY – Message buttons send messages from the TeensyMaestro memories.

FLEX – Message buttons send messages from the Flex CWX memories.

Messages sent by selecting them from the CW Message Menu always send the TeensyMaestro messages, regardless of the source.

* + 1. CW Paddles

Right Handed

Left Handed

Default is Right Handed, can be changed in config file.

* + 1. CW Mode

A – Iambic mode A.

B – Iambic mode B.

U – Ultimatic (last element repeats, has dot and dash insertion).

Default is B, can be changed in config file.

* + 1. Set contest serial number:

Select this option with a short press of the Menu Select button. The highlight will move from the entire line to the current setting. Now the CW Speed knob can be used to adjust the serial number. Short press the Menu Select button to accept the number, save it to EEPROM and exit the menu. Long press the Menu Select button to exit without changing the value.

* + 1. Clear contest serial number to 1

Sets the serial number to one. Typically done at the beginning of a contest.

* 1. CW Message Menu

List of CW messages from SD Card. Each message can be played from the menu or from the 6 message buttons (see CW Message Source, above). A tap of the key will stop any message in progress.

* + 1. CW Message Commands

$C – Send MyCall. MyCall is set in the config file.

$N – Send next serial number. Increment serial number by one after the send.

$NR – Send last used serial number. Do not increment serial number. Useful to provide a fill if the other station requests it.

$N+ – Increment the serial number without sending it. Useful as a stand-alone key (i.e. a CW message with no other text or commands in it) to increment the serial number if the previous contact was sent manually.

$N- – Decrement the serial number without sending it. Useful as a stand-alone key to decrement the serial number if it was accidently increased or in case of a dupe.

$Snnn – Change CW speed to nnn WPM, return to speed prior to $S command when message finishes playing.

$SPnnn – Change CW speed to nnn WPM, leave this speed in effect when message finishes playing.

$S – Change speed back to speed prior to $S command executed in same message.

$R – Repeat message continuously. Tap the key to stop message.

$Pnnnn – Pause nnnn milliseconds. Generally used with the $R command to listen for calls between messages or to time beacon transmissions.

\*xx… – Prosign. Any number of characters following the \* will be run together to form a prosing (e.g. \*AS for wait). A space or end of line ends the prosign.

\_ – Cut zero. Underscores will be sent at 1.5 times the length of a normal dash.

1. Configuration File
   1. SD Card Configuration File

The **MMConfig.ini** file provides a way to customize the functionality of the TeensyMaestro and must be on the SD Card and named exactly like this to work. If there is no config file on the card or if the card is not in its slot when the TeensyMaestro is started, all values will take on defaults, but the unit will operate. Some functions will be limited or not useful, for example CW message memories will be blank.

The SD Card is read at power up, and the values are parsed and stored internally. The SD Card can be inserted or removed while the TeensyMaestro is powered up. This provides a method to make changes without re-starting the TeensyMaestro. To apply changes made while the unit is powered up, use the “Reload Config File” menu option.

* 1. Editing MMConfig.ini

This file can be edited in any text editor like Windows Notepad. Lines need to end in CR (0xA) or CRLF (0xA 0xD) for the file to function correctly. Notepad takes care of this automatically, so it is a good choice to use.

* 1. Configuration settings and defaults
     1. Configuration parameter syntax

Configuration parameters have three parts; the parameter name, a colon followed by a space and data. Each setting can be in upper, lower or mixed case. All commands and data will be converted to upper case for configuration parsing.

* + 1. Comments

Comments are preceded with a semicolon “;”. A comment can start at the beginning of a line, in which case the whole line is ignored, even if it contains otherwise legal configuration settings. Comments in the body of a line cause the rest of the line to be ignored.

For example, this line is ignored because of the leading semicolon:

;WPM: 24 ; Sets default WPM. Comment out to start with last used WPM.

If the leading semicolon is removed, the “WPM: 24” configuration will be applied and everything from the second semicolon to the end of the line will be ignored.

* + 1. MyCall:

Freeform text. Will be used on the splash screen and is the text that will be sent if the $C macro is used in a CW memory. No default, no checking for legitimate callsigns.

* + 1. MyLicense:

Can be Extra, Advanced, General, Technician, Tech or Novice. Used to determine in band operation. Default is Extra. Tech and Technician are equivalent.

* + 1. Splash:

Time to leave the splash screen up in milliseconds (e.g. 10000 = 10 seconds). Default is 0.

* + 1. Screen Save:

Screen saver time out in seconds. The screen saver kicks in after the specified time out period has elapsed since the last button press, knob turn, or display update (except for the S meter). The screen saver is implemented as a blank screen. Default is 1200 (20 minutes).

* + 1. Power Button:

Configures the power button to power the TeensyMaestro on or off or act like a reset button. Valid values are Power and Reset. Default is Power.

* + 1. WPM:

Keyer speed (5 – 100 WPM). No default (uses CWX keyer speed from the Flex).

* + 1. VFO A SSB Default Rate:

Number of VFO A pulses between encoder updates in SSB mode. The lower the number, the faster the tuning rate. If it is too small, it gets to be ridiculously difficult to set the VFO to a particular frequency. Can only be changed in the MMConfig.ini file. Default is 40.

* + 1. VFO B SSB Default Rate:

Number of VFO B pulses between encoder updates in SSB mode. Default is 40.

* + 1. VFO A CW Default Rate:

Number of VFO A pulses between encoder updates in CW mode. Default is 10.

* + 1. VFO B CW Default Rate:

Number of VFO B pulses between encoder updates in CW mode. Default is 10.

* + 1. VFO A SSB Default Step:

Sets the default step size for VFO A in SSB mode. Step size can be changed from the front panel. Note that steps are specified using a step index of 0 – 7. Default is 3 (100 Hz).

0 = 1 Hz step

1 = 10 Hz step

2 = 50 Hz step

3 = 100 Hz step

4 = 500 Hz step

5 = 1000 Hz step

6 = 2000 Hz step

7 = 3000 Hz step

* + 1. VFO B SSB Default Step:

Sets the default step size for VFO B in SSB mode. Step size can be changed from the front panel. Note that steps are specified using a step index of 0 – 7. Default is 3 (100 Hz).

* + 1. VFO A CW Default Step:

Sets the default step size for VFO A in CW mode. Step size can be changed from the front panel. Note that steps are specified using a step index of 0 – 7. Default is 1 (10 Hz).

* + 1. VFO B CW Default Step:

Sets the default step size for VFO B in CW mode. Step size can be changed from the front panel. Note that steps are specified using a step index of 0 – 7. Default is 1 (10 Hz).

* + 1. Long Press:

Time in milliseconds for a button press to be considered a long press. Default is 250.

* + 1. Vol A encoder steps:

Number of encoder increments per detent click. If you turn the volume encoder knob for slice A and get more than one value increment per click, then you need to set this to the number of jumps in value per click. Most encoders will be 1:1, 2:1 or 4:1. If all encoders are the same, change all of the encoder step values to the same value. If you have a mix, set each as needed. Default is 1 (1:1).

* + 1. Vol B encoder steps:

Number of encoder increments per detent click.

* + 1. AGC A encoder steps:

Number of encoder increments per detent click.

* + 1. AGC B encoder steps:

Number of encoder increments per detent click.

* + 1. Low A encoder steps:

Number of encoder increments per detent click.

* + 1. Low B encoder steps:

Number of encoder increments per detent click.

* + 1. High A encoder steps:

Number of encoder increments per detent click.

* + 1. High B encoder steps:

Number of encoder increments per detent click.

* + 1. CW encoder steps:

Number of encoder increments per detent click.

* + 1. Profile: <prof name>

Startup profile. Sets the radio to this profile whenever the TeensyMaestro is powered up. The profile name must be placed within angle brackets, and must be an existing profile in the radio. Invalid profiles will be ignored. Comment out to leave the radio on the last used band/profile. No default.

* + 1. CW Msg Source:

Determines whether the CW message buttons send the messages from the SD Card in the Teensy or from those stored in CWX in the Flex radio. Values are Teensy or Flex. Default is Teensy.

* + 1. CW Paddles:

Values are Left or Right for left or right handed operation. Default is Right.

* + 1. CW Mode:

Values are A, B or U for iambic mode A, iambic mode B or Ultimatic. Default is B.

* + 1. CW Msg:

Freeform text to be sent when selected from the CW Message menu (regardless of value of CW Msg Source) or when the CW Message buttons are pressed (if CW Msg Source = Teensy). Up to twelve messages can be defined.

* + 1. TeensyIP:

IP address to use when connecting directly to the radio (i.e. no DHCP server). Default is 169.254.0.25. If Flex, computer and TeensyMaestro are connected directly using a network switch, change this address to be on the same network segment. If it does not work using the default, go to CMD mode on the computer (cmd.exe) and enter ipconfig. The computer IP address will show as **IPv4 Address. . . . : nnn.nnn.nnn.nnn**  Duplicate the first two numbers and make sure that the whole address is not a duplicate of any other device on the network.

* + 1. TeensyGateway:

IP gateway address to use when connecting directly to the radio (i.e. no DHCP server). Default is 169.254.0.1.

* + 1. TeensyMask:

IP mask to use when connecting directly to the radio (i.e. no DHCP server). Default is 255.255.0.0.

* + 1. Out Of Band Indicator:

Values are ON or OFF. Used to send a spot to the SmartSDR screen (requires spots to be enabled in SSDR) indicating out of band for the MyLicense class. Also turns the frequency and license class red in the TeensyMaestro display. Default is ON.

* + 1. Out Of Band Spot Time:

Number of seconds to leave an out of band spot on the SmartSDR screen. Default is 5.

* + 1. 160 CW:

Five pairs of frequencies representing the CW privileges for Extra, Advanced, General, Technician and Novice class licensees respectively. Each value is separated from the next by a comma. Zero values indicate no privileges. Works in conjunction with Out Of Band Indicator and Out Of Band Spot Time (above). Default is privileges as of December, 2020.

* + 1. 160 PHONE:

Five pairs of frequencies representing the phone privileges for Extra, Advanced, General, Technician and Novice class licensees. Each value is separated from the next by a comma. Zero values indicate no privileges. Default is privileges as of December, 2020.

* + 1. 80 CW:
    2. 80 PHONE:
    3. 40 CW:
    4. 40 PHONE:
    5. 30 CW:
    6. 30 PHONE:
    7. 20 CW:
    8. 20 PHONE:
    9. 17 CW:
    10. 17 PHONE:
    11. 15 CW:
    12. 15 PHONE:
    13. 12 CW:
    14. 12 PHONE:
    15. 10 CW:
    16. 10 PHONE:
    17. 6 CW:
    18. 6 PHONE:
    19. Display SPI Clock:

Sets the TFT display SPI clock frequency in MHz. Default is 20. If the display does not start up correctly every time, change this value up or down until you find a frequency that works.

1. Troubleshooting
   1. Dreaded white screen of death.

This may manifest itself as a snowy or otherwise corrupt screen. Most likely cause is SPI Clock speed. Find this line in the MMConfig.ini file on the SD card:

**Display SPI Clock: 30 ; Default = 20 change this up or down if you get a corrupt display**

Start by changing the value to 20. You will need to power the TeensyMaestro off and back on to use the new value. If that does not resolve the issue, try a value of 10. If it won’t work at 10, there is likely a wiring problem between the Teensy board and the TFT display.

The Teensy board communicates with the TFT display using Serial Peripheral Interface (SPI) operating at clock speeds in the tens of MHz. Long signal lines will tend to reduce the maximum clock frequency that can be used. Because the clock is operating at RF, reflections on the lines can sometimes be better at higher frequencies, sometimes better at lower. This means that you may need to experiment some to get this working properly.

* 1. Can’t load firmware using Teensy.exe.

If you hover your mouse over the file name in the Teensy.exe panel, you should the name of the file and a percentage usage. If, instead of the percentage it says “(too large!)”, then you likely do not have a good USB cable. Check to be sure that you are not using a “charge only” cable.



* 1. Both slices show blank, no controls are responsive except for CW Speed.

Check to be sure that the TeensyMaestro is connected to the same LAN segment as the Flex radio. Power the TeensyMaestro off and back on to re-try.

* 1. TeensyMaestro connects, then Flex radio crashes with a software error (power button double-flashes red, then restarts).

As of December, 2020, this is a known issue with the Flex API and is supposed be fixed in the next release.

This happens when you connect to the Flex radio without SmartSDR running. Normally, you should start SmartSDR, then power up the TeensyMaestro. To use the TeensyMaestro without SmartSDR (stand-alone mode), press and hold any button while powering up. The TeensyMaestro will let you know when to release the button.

Many SmartSDR features like band change and antenna selection are not available in stand-alone mode. The TeensyMaestro is designed to work *with* SmartSDR, not instead of it.

If SmartSDR is started after starting the TeensyMaestro in stand-alone mode, it will use multiFLEX to connect (V3.xx and up), and you will not have true parallel operation of SmartSDR with the TeensyMaestro. SmartSDR versions lower than 3 will not work in this mode.

* 1. TeensyMaestro keyer produces all dots.



Turn off Iambic when using the TeensyMaestro keyer if it is plugged into the keyer jack or you will get all dots. If you connect the keyer to pin 4 of the 15 pin accessory connector, Iambic can be on or off to control the behavior of whatever you have plugged into the key jack and will not impact anything hooked up to the accessory connector.