Deconstructing Deviation Model files

A Practical Guide to Using Deviation Models for New Users

By Jeffrey Diehl

Contents

Deconstructing Deviation Model Files

Introduction

IMPORTANT: Please review the Deviation disclaimer and copyright notices at the end of this document.

The Model.ini files are an integral part of the Deviation system. Most deviation users can download a model.ini file that matches the model they want, and use it with their model straight away, but as manufacturers produce new models, the model.ini files may need to be adjusted. The internal mechanisms used in the model.ini file are not terribly complex, but they can be intimidating for many users, so this tutorial is written to help demystify these files.

I've tried to focus the model.ini material on features that seem to be attractive to most users, and there are other ways to gain functionality or features. Deviation 4.0.1 is the most recent "final" release of this system, however subsequent "nightly" updates add protocols and functionality beyond that release. This tutorial assumes that the user has already installed the desired version and that the required hardware is installed and working.

This tutorial relies heavily on the User Guides located on DeviationTX.com. You MUST read the appropriate user guide for your transmitter prior to attempting any changes to model.ini files. Make yourself familiar with chapters 8 and 9 and you'll be an expert in no time at all.

Following the installation of the Deviation system, the user can edit the model setup using the built-in (GUI) model setup on the transmitter, and we'll follow that organizational model in this discussion. The user can also access to the file system through a USB connection, over which they can copy, edit and install model setup files (we call them model.ini files), icons, or other customized files. The user can install files from other users, or edit the files directly on the PC or MAC.

> *Tip: The Windows "Notebook.exe" file will corrupt Deviation files, because it doesn't respect line endings correctly. Other programs, like "Notebook++.exe" will edit these files correctly.*

Deviation also provides an emulator program that runs under Windows. This emulator matches the functionality of the transmitter GUI, and provides an excellent system for developing and testing model.ini files. The file structure of the emulator is an exact model of the one in the transmitter, so moving files between the PC and radio is simply a copy/paste over the USB connection.

Note: In general, the best model.ini files are the result of editing on YOUR Devo transmitter or the related emulator, using the GUI screens. If you edit them manually on your PC or MAC, you may make errors that the Devo firmware cannot understand. However there are some functions that are much easier to edit or copy (such as sticky throttle or common rates); I recommend trying them in the emulator to ensure functionality and safety.

The most valuable resource is the community of Deviation developers and users. Their software, questions, answers, and tutorials are located at DeviationTX.com. I am deeply grateful for their efforts and support.

Deviation Model Setup Tools

Create a new model

Using the Deviation menu on the transmitter, the first option is to use Model Setup tools to create and edit a new model. As the model is set up, those setting are saved in a Model.ini file in the Models folder in Flash memory. Most users create a new basic model.ini using the transmitter GUI, after which the user can connect the USB and edit the file on a PC or MAC.

An initial 30 Deviation model files are installed into the Models folder upon installation of the Deviation system, named, appropriately, "model1.ini", "model2.ini", "model3.ini", up to "model30.ini". Each file can work independently of any other, and contains all of the definitions for transmitter protocols, stick and switch assignments, display arrangement and a variety of customizable options. For convenience, we'll use "model.ini" in the generic form for our examples.

> *Tip: You can add more than the 30 standard model definitions! Simply make a copy of the standard model.ini file and copy it to the Models folder, continuing the naming scheme beginning with "model31.ini" up to "model99.ini".*

Overview of the sections

A model.ini file contains several entries, starting with the name of the model and the mixer mode used for this model. This is followed by sections designated by "[*section name*]" in brackets, and each section contains the definitions and customization needed by this model. This includes the radio module type, protocol and protocol options, channel assignments, mixer components, trims, timers, telemetry and GUI display layout options. Most files are about 4k bytes long, and you'll often see several "@" characters at the end of the file; these are artifacts created by the way the Deviation file system and can be ignored or erased if desired.

Like ALL Deviation files, the model.ini file MUST be edited with an editor that respects line ending characters. The most popular editor seems to be "Notepad++" and I find it to be an exceptionally well done editor. DO NOT use the Windows Notepad.exe editor, as it will not respect these special characters and will leave the file unusable by the transmitter.

> *Tip: The file "default.ini" in your "Models" folder is used to create the channels and trims for each new model, so if you have sections or custom items you want in every model, update the "default.ini" file.*

Each section of the model.ini file contains very detailed information about that section, and the syntax and options are well described in the online Deviation User Guide for each transmitter. With the variety of transmitter radio modules, channel assignments and other options available, these User Guides offer an in-depth view at a very detailed level. By contrast, this discussion will just touch upon some of the most common model options.

There have been a few discussions about adding comments or documentation lines to the model.ini files, but for now the files do NOT contain any such text. This is driven by the fact that the transmitters are limited in memory, and a desire to keep a common file and syntax structure across the Walkera Devention transmitter line.

Required Model.ini Sections

Every model.ini file must contain sections denoted by a section name inside brackets ("[channel1]"). There are several sections, beginning with an unnamed section with the model name and model mode statement followed by the radio/protocol information. This is followed by several sections with channel information, trim, and GUI display information. A simple model for a three-channel airplane might be only about 100 lines of text. A more complex model, with more sophisticated features like Sticky Throttle Hold, Camera features, multiple flight modes, and so on, might have a couple hundred lines. Each section is usually pretty small, and for most users, even the newest ones, the details of each section are clear in the first or second reading.

> *Tip: The first time you set up a new model, create the model using the Graphic User Interface (GUI) tools on the transmitter. This will create a model.ini file that contains all the right sections for your model. You can later customize the model.ini file to your own tastes using an editor via the USB connection.*

Header – Model Menu -> Model Setup

The header for each model.ini file does not have a section name. It contains the model name, mixer model and type of craft (which may affect how the channels are mixed). For this discussion all of the model.ini files reviewed are created with mixermode=Advanced.

> *Tip: If you change the settings in these first two sections, other sections will change or be reset to defaults!*

Section Name	Section Text	Comments
(None)		
	name=Model7	This is the name that shows up on your display.
	mixermode=Advanced	The Advanced mixer mode is easiest to start with. Standard mixer mode is used with Helicopters to directly manage features, usually by more experienced users. If you change between modes, many other sections will be affected!
	type=multi	The three types are "heli", "multi" and "plane". These selections may affect the channel output, so choose the appropriate type.

Radio Protocol

[radio]		This section describes the specific radio communications details for each protocol.
	protocol=YD717	Protocols are described in detail in the User Guide. If you change protocol, many other options may change!
	num_channels=5	This number is the number of channels desired to control the model. It limits the number of channels you can edit on the TX

		GUI. It does not limit virtual channels and additional channels can be added in this file.
	fixed_id=123456	Use your own fixed ID; this one is an example, and should be changed
	tx_power=150mw	150mW is the maximum power; you want to adjust this to your use. For example, indoor flyers can probably use 10mW
[protocol_opts]		This section will apply to many, but not all, protocols. The options vary by protocol.
	Format=Sky Wlkr	Protocol options can be accessed in the GUI by clicking on the protocol name.

Channels – Model menu -> Mixer

Each channel has a separate section, so there is one such section for all channels listed in the above "radio" section. However, you may have additional channels sections (for example, a model with num_channels=5 can still have a section for channel 6 or more. These extra channels can be used to improve display options, and they work just like to virtual channels, except that they cannot be directly edited by the GUI unless you set num_channels to include them (you can then reset the num_channels back down and not lose these sections.

> *Tip: Most protocols use the channels in a "AETR" order – channel 1 is Aileron, channel 2 is Elevator, channel 3 is Throttle, and channel 4 is Rudder. There are exceptions: DSMX and DSM2 as well as Walkera and KN protocols use TAER. If you change model type or GUI type, the channel order may change, too!*

Simple template, fixed curve, no expo

This template simply sets the aileron stick on the TX to send a value from -100 to 100 to the channel 1, which is the Aileron on the aircraft. This is common with small helis and planes.

[channel1]		Each channel will have a separate section
	template=simple	This is the simple template which allows just one mixer setting
[mixer]		
	src=AIL	This mixer uses the Aileron stick for input
	dest=Ch1	In this model, channel 1 is the Aileron

Expo Template with multiple Rates

This is a simplified approach to setting up multiple servo throw rates on a channel. For pilots starting out with a new aircraft, this model keeps the pilot from over-controlling the aircraft while they learn the aircraft flight characteristics.

In this setting, the aileron will be at 60% throw when switch B is in the 0 (default) position, 80% when switch B is in the 1 position, and 100% (default) when switch B is in the 2 position. The stick value is

modified slightly with an “Expo” curve deflection setting of -10 which makes the stick less sensitive around the middle values. By default the output value of the first mixer is replaced by the output value of the second mixer, although you can also add or multiply. More options are available (see the user guides for further details).

[channel1]		
	reverse=1	This ensures that all mixers use reverse input
	template=expo_dr	Use the Expo_DR template so we can easily use Expo points and up to three data rates
[mixer]		First mixer setting
	src=AIL	This mixer uses the Aileron stick for input
	dest=Ch1	In this model, channel 1 is the Aileron
	scalar=60	Scale the stick input by 60%
	curvetype=expo	Use an expo curve
	points=-10,-10	The expo curve deflection points
[mixer]		Second mixer setting
	src=AIL	This mixer uses the Aileron stick for input
	dest=Ch1	In this model, channel 1 is the Aileron
	switch=SW B1	Use this mixer when switch B is in position 1
	scalar=80	Scale the stick input by 80%
	curvetype=expo	Use an expo curve
	points=-10,-10	The expo curve deflection points
[mixer]		Second mixer setting
	src=AIL	This mixer uses the Aileron stick for input
	dest=Ch1	In this model, channel 1 is the Aileron
	switch=SW B2	Use this mixer when switch B is in position 2
	curvetype=expo	Use an expo curve
	points=-10,-10	The expo curve deflection points

Complex Expo with Multiple Rates

This mixer is set up for a v202 protocol quad with LED control on Channel 5. Some of these models will set the flash rate from the lowest rate (off) when the switch is -100, to the highest rate (fully on) when the switch is 100. So this mixer can be used make sure the LEDs flash quickly (value of 99) when the quad is in flip mode.

Since the curve is not explicitly declared, the default curve is a 1-to-1 curve, straight from -100 to 100. When SW A goes to position 1, the output value will be 100 on channel 5, which puts the lights on full. When switch FMODE is set to 1 (which in channel 6 sets flip mode), the mixer sets a value of 99 on channel 5 which causes the LED to flash very quickly.

[channel5]		
	template=complex	Use the complex mixer
[mixer]		First mixer setting
	src=SW A1	This mixer uses the value of switch A1 – note this is different than using switch= command which would determine when the mixer is active – see below
	dest=Ch5	In this model, channel 5 turns on LEDs
	usetrim=0	Don't apply any trim values
[mixer]		Second mixer setting
	src=FMODE1	Use the value of the FMODE1
	dest=Ch5	In this model, channel 5 turns on LEDs
	switch=FMODE1	This mixer is active when FMODE is on (e.g. 100)
	scalar=99	Apply 99% to the switch value (e.g. 99)
	usetrim=0	Don't apply any trim values
	curvetype=fixed	Use a fixed value (of 99) and replace the output from the previous mixer

Graduate work: Using Scalar to set a value

The following channel 7 mixer shows a common 3-way switch mixer. This sort of switch may be used to set three data rates, flight modes, or other use depending on how your receiver or flight controller uses a 0%, 50% and 100% value.

This can be applied to any 3-way switch, which may vary by transmitter model: just substitute the "src=" and "switch=" name for the appropriate switch. By default, a physical switch has a software value of 100 (e.g. it's "on), at each position, so when set at position 0, SW A0 has 100, at position 1, SW A1 has 100, and position 2, SW A2 has 100. By applying the right scalar, this mixer set the output values to ensure that the mixer results in -100 for position 0, 0 for position 1 and 100 for position 2. Remember, each mixer is by default replacing the output value of the previous mixer.

> *Tip: For higher performance you can set the scalar to more than 100%, so on a model with a slow rudder, I might set the rudder scalar=125 to improve the rudder response.*

A common mixer for a 3-way switch (down-middle-up)

[channel7]		
	template=complex	
[mixer]		The first mixer sets up the defaults for this switch
	src=SW A0	Use the value of switch SW A position 0
	dest=Ch7	This channel could be any channel
	switch=SW A0	The mixer will be activated when SW A is in position 0
	scalar=-100	Since position 0 has a value of 100, use a scalar of -100 to send a switch value of -100
	curvetype=fixed	Use "Fixed" to ensure that the value at each position is always the chosen value
[mixer]		
	src=SW A1	Use the value of SW A position 1 (middle position)
	dest=Ch7	
	switch=SW A1	The mixer will be activated when SW A is in position 1
	scalar=0	Since position 1 now has a value of 100, use a scalar of 0 to send a switch value of 0
	curvetype=fixed	Use "Fixed" to ensure that the value at each position is always the chosen value
[mixer]		
	src=SW A2	When the switch is moved to position 2 (down position)
	dest=Ch7	

	switch=SW A2	The mixer will be activated when SW A is in position 2
	curvetype=fixed	Since position 2 now has a value of 100, you don't need a scalar! Use "Fixed" to ensure that the value for this position is always 100.

Safety Check – Model Setup -> Mixer (click on channel name)

An often-overlooked section of the model.ini file is the Safety check. This function checks to ensure that the throttle output is set to where you want it – usually the minimum value or 0 - before the TX turns on the protocol. For most uses, the "Auto=min" is great; this simply checks to see if the throttle output based on your current stick position is set to the minimum value, and if not, you'll get a warning screen. Using "Auto" tells Deviation to guess your throttle channel, so if you're getting a warning when your stick is already at a minimum setting, try changing this to the known channel, "Ch3=min", or "Ch1=min".

Note that if your model flight mode needs to have the throttle in the middle (for example, collective pitch or "idle-up" helicopters), "Auto=min" will show a warning, and you would need to change flight mode to clear the warning message.

You can also set this to "Auto=Zero" to check for a zero output value or "Auto=Max" to check for a maximum value. Please see the user guide for details.

Typical Safety Check setting

[safety]		
	Auto=min	Check to see if the throttle is at the minimum position.

Trims – Model menu -> Trims

The trim buttons, like any buttons in Deviation, can be used as normal stick input trims to center your sticks, if needed. And they can also be used as general purpose buttons in several ways. They can be step buttons, that increase in value when you press "+" or decrease when you press "-" (the normal trim function); momentary buttons you simply press to turn on, and release to turn off (like taking a snapshot photo); toggle buttons that turn on when you press them, and stay on until you press them again (perhaps to turn on a video camera); or on/off buttons, so when you press the "+" it turns the switch on, and when you press the "-" it turns the switch off.

Since a trim button basically adds a value to a mixer, you can either use a physical stick (THR, AIL, ELE or RUD) or switch, or you can set up a virtual switch. For normal trim use, you don't need to specify the switch, but the other applications need a switch on which to apply their action.

Using the trims in the normal fashion to trim rudder

[trim3]		
	src=LEFT_H	Source is the left horizontal trim – defaults to RUD unless you change it
	pos=TRIMLH+	Pressing to the "+" position gives positive trim
	neg=TRIMLH-	Pressing to the "-" position gives negative trim
	value=-1,0,0	Use an initial value of -1 (note trims are saved when you use them!). If you set this to 5 you'll get more trim faster. There are three parameters because this can be used for three-position switches.

Using a trim as momentary switch

While not terribly complicated, using a trim as a momentary switch requires three components: first you need to know the output channel for the button; we'll use channel 8 which is often assigned to snap a photo on the camera. You also need to have a virtual channel to which you will assign the trim (and you can name a virtual channel). Last you need to set up the trim button(s) and values.

[channel8]		This mixer will send an "On" value when it is activated by the left horizontal trim ("-") button, and cause the camera snap a still photo
	template=simple	Keeping it simple
[mixer]		
	src=Virt2	Use the value of Virt2; if Virt2 goes to 100%, this mixer will go to max.
	dest=Ch8	Send the mixer results on channel 8
	curvetype=zero/max	Use a curve that goes from 0 to 100 (or max)
	points=0	No expo points
[virtchan2]		Set up Virt2 normally at 0
	name=photo	In the GUI on the transmitter you'll see "photo" as the channel name
	template=complex	
[mixer]		
	src=AIL	With this mixer, src=Ail comes up when we don't need a source value. The GUI will show "None".
	dest=Virt2	Keep the results value on this switch
	scalar=0	Set the scale to 0, meaning normal value is 0
	curvetype=fixed	This keeps the value at 0 until the trim is actuated.
[trim5]		Trim 5 will set the left vertical trim "-" key to momentarily actuate the Virt2 switch (that is, turn on the switch).
	src=Virt2	Use the current value of Virt2 (is normally 0)
	pos=TRIMLV-	Use the left vertical "-" button
	step=193	Step 193 is a special number that indicates that this trim is set to a momentary button
	value=-100,0,0	Set the initial trim at -100

Using the trims as toggle switches

Ok, so there is a more complicated way to do just a bit more, and if you're with me so far, I'll push the limits! This trim setup uses a momentary trim to turn the video camera on and off.

As I fly I may not be able to see if the camera is on or off on the aircraft, so I also use a toggle trim to display the camera icon.

The complicated part is keeping the camera toggle icon visible when the video is turned on. This requires another trim, which uses the same button that turns on/off the video. It also needs another virtual switch so we can assign the button value. Lastly, you need to assign a toggle icon to a channel in the display layout "[GUI]" like this: Displays camera status

The resulting mixer set includes both a momentary trim (Virt1), and a toggle trim (Virt3).

[channel7]		Set up the channel assignment first
	template=simple	This mixer will send an "On" value when it is activated by the trim button, and cause the camera to start or stop recording
[mixer]		
	src=Virt1	Use virtual 1 as a source value (see below)
	dest=Ch7	The video function in this protocol uses channel 7
	curvetype=zero/max	Set the curve to be 0 when inactive, and max (e.g. 100 or "On") when the mixer is active
	points=0	(No expo points are used in this curve)
[channel10]		Use channel 10 for the GUI Icon toggle. Note for video we want to keep it on the screen while recording, so Virt3 handles that for us
	template=simple	
[mixer]		
	src=Virt3	Use Virt3 to tell us if the camera is on (100) or off (0).
	dest=Ch10	Channel 10 isn't actually used by this protocol, but the display GUI needs it for the toggle.
[virtchan1]		Now add a virtual channel for the trim
	Name=video	In the GUI on the transmitter you'll see "video" as the channel name
	template=complex	In a trim, this is used for a momentary switch
[mixer]		
	src=AIL	This means don't use a source in this case
	dest=Virt1	The destination output mixer is Virtual 1
	scalar=0	Use a scalar of 0 to keep the value at 0 until pressed
	curvetype=fixed	Keep this value at 0

[virtchan3]		Set up another virtual channel to allow the still photos to toggle the GUI display icon briefly.
	Name=video2	In the GUI on the transmitter you'll see "video2" as the channel name
	template=simple	
[mixer]		
	src=Virt2	When the still photos are taken, Virt2 is active for a short time, so we pass that through Virt3 to briefly show the camera on the GUI display.
	dest=Virt3	Keep the value on this mixer
	curvetype=fixed	Use a fixed value curve
[trim1]		Trim 1 is a momentary trim setting trim 1 to the left vertical "+" key for video. When pressed it will trigger the actions in virtual channel 1
	src=Virt1	Use virtual channel 1; in the GUI you will see "video" as the channel name
	pos=TRIMLV+	Mixer is active when the left vertical trim is pressed to the "+"
	step=193	Step 193 is a special number that indicates that this trim is set to a momentary button
	value=-100,0,0	Start at -100 ("off")
[trim3]		Trim 3 is a toggle trim that also responds to a press of the left vertical "+" key (e.g. start or stop video). This toggle trim inverts the value of Virt3 (starts out off, so off to on, then later on to off the next time the "+" key is pressed). This way the icon is only visible when the camera is on.
	src=Virt3	Use the current value of virt3
	pos=TRIMLV+	Use the left vertical "+" button
	step=192	Step 192 is a special number that indicates that this trim is set to toggle the value
	value=-100,0,0	Start at -100 ("off")

Using the trims as on/off switches

The last trim approach is called On/Off. The + side of the trim switch turns it on, and the - value turns it off. Channel 5 is used for lights in many protocols. Lights usually start out on, so we'll use the reversed values on channel 5. We'll also set up a virtual channel to hold the values, and then use that virtual channel as our source for the trim.

[channel5]		Channel 5 often controls lights on/off, so we'll use an on/off trim
	Reverse=1	The lights should be ON to start
	template=simple	Keep it simple!
[mixer]		
	src=Virt5	Use the value of Virtual 5
	dest=Ch5	Send the output results to Channel 5
	scalar=0	Scalar of 0 …
	curvetype=min/max	Curve type will go from -100 to 100
	points=0	No expo points
[virtchan5]		Use this virtual channel
	name=Virt5lt	Name it Virt5LT so we know it's the lights
	template=complex	Choose a complex mixer
[mixer]		
	src=AIL	Default for no source channel
	dest=Virt5	Send the results to Virt5
	usetrim=0	Don't apply any trim to this result
	curvetype=fixed	Use a fixed curve from -100 to 100
[trim3]		
	src=Virt5	Use the current value of Virt5
	pos=TRIMLH+	To set the switch on use TRIMLH+
	neg=TRIMLH-	To set the switch off use TRIMLH-
	step=191	Step 191 is a special number that indicates that this trim is set to use both directions to turn on and off
	value=-100,0,0	Start at -100 ("off")

Timers – Model menu -> Timers

One of the most popular features of the Deviation system is the ability to time your flights. Deviation allows up to four timers, and most screen configurations on the Devo 7e use three of those timers, one to count-down the time of the flight – this helps prevent over-discharging your batteries – then one to count-up the elapsed time for this flight, and finally, one to keep track of all the time the model has been active.

> *Tip: You set the time for the countdown timer to the expected battery discharge time in seconds. For many of my small 1S models, I find they run about 4 to 5 minutes before the model's low voltage indicator goes on (usually about 3.2v under load), and a "time=" number of 240 to 300 (in seconds) is where I start. I usually tune this time so I avoid running my batteries too low.*

Typical timer setup

[timer1]		
	type=countdown	Counts down to 0 from our start time
	src=Ch3	Count only when channel 3 ouput value is greater than zero (e.g. the throttle in this case)
	time=330	Number of seconds to count
[timer2]		
	src=Ch3	Simply counts elapsed time whenever channel 3 output is greater than 0,
[timer4]		
	type=permanent	Keep a running total for this model.ini
	src=Ch3	Count elapsed time whenever channel 3 output is greater than 0.
	val=9670136	The current elapsed time in milliseconds

Telemetry Alarms – Model menu -> Telemetry

Advanced users will use telemetry to track the model's real-time status. Some small models provide battery voltage, and more complex models will provide other important data. The telemetry provided is also dependent on the radio protocol. Deviation developers have incorporated telemetry into a number of protocols where possible, and more are in development. Consult DeviationTX.com for details on telemetry options for each protocol.

> *Tip: Enable or disable telemetry for a protocol in the Model Setup menu by clicking on the protocol name. If telemetry is available, you can turn it off or on.*

Typical Hubsan Voltage Telemetry

[telemalarm1]		
	source=Volt1	Volt1 is the battery voltage for the Hubsan protocol
	above=1	0 is ">", 1 is "<="
	value=32	Tenths of volts, so 32 is 3.2volts

User Display Layout – Model menu -> Main Page Config

The default display layout is dependent on the type of Devo TX you have. The Devo 7e and Devo 10 have a Graphical User Interface of 128x64 pixels. The Devo6S and Devo8S have a GUI of 320x240 pixels in color. Other transmitters will have other size displays. The Layout folder contains a "default.ini" file with the layouts for each model.

> *Tip: The file "default.ini" in the "Layout" folder is used to create the screen display for each new file, so you can customize that file for your own tastes.*

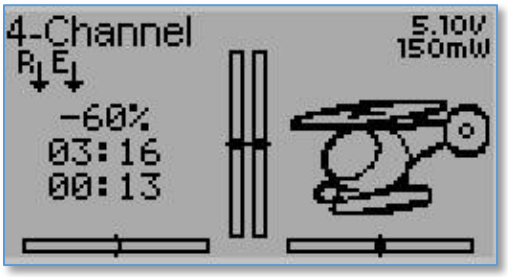
The screen consists of trim bars, big- and small-boxes, toggles and model icons as well as other objects. On models except the Devo 7e, you can set the relative position of each object in the layout screen; on the Devo 7e, you can change the column and row position with an editor on your PC.

This section allows you to use icons to represent the position of each switch, which are called Toggles. On all models, you can click on the Toggle button to choose the icons you wish to assign to a given toggle.

> *Tip: Use the Toggle3.bmp file to add your own toggle icons. The standard 4.01 version includes a basic set of icons to which you can add more of your own, or you can download a toggle3.bmp from deviationtx.com. The examples here for a Devo7e use an expanded toggle3.bmp published in 4/2015.*

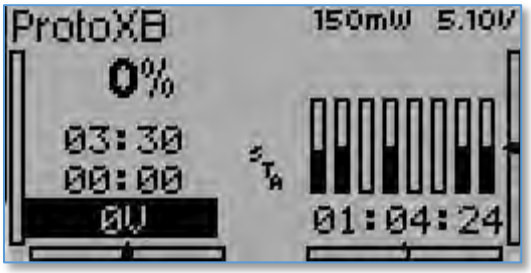
> *Tip: The Quickpage entries allow you to use just one navigation key to quickly get to specific GUI pages, such as the Model Setup page, Telemetry display, Channel monitor, etc. By default you can access these pages (when viewing the main display page) by long-pressing the Up+ key, and pressing it repeatedly will cycle through the pages.*

Standard 4-channel Heli displaying throttle position (-100 to 100), model icon and timers



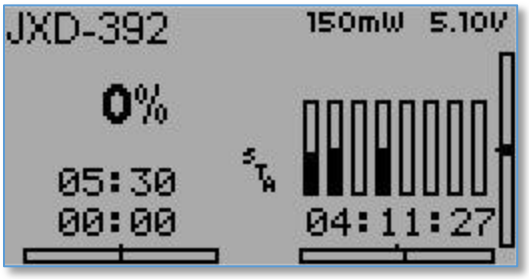
[gui-128x64]		
	V-trim=59,10,1	Displays the throttle trim
	H-trim=5,59,3	Displays the aileron trim
	V-trim=65,10,2	Displays the elevator trim
	H-trim=74,59,4	Displays the rudder trim
	Small-box=2,22,Ch3	Display the throttle output
	Small-box=2,31,Timer1	Countdown timer with sticky hold
	Small-box=2,40,Timer2	Stopwatch timer with sticky hold
	Model=75,20	Position the model icon here
	Battery=102,1	Display the TX battery voltage
	Toggle=4,10,0,3,0,HOLD	This toggle icon will be located at column 4, row 10 of the 128x64 matrix. Three icons can be used, but only the center icon is defined. It is assigned to the HOLD switch
	Toggle=13,10,0,5,0,FMODE	This toggle is assigned to the FMODE switch
	Toggle=22,10,0,4,0,None	This toggle is not used (“none”)
	Toggle=31,10,0,0,0,None	This toggle is not used (“none”)
	Toggle=40,10,0,0,0,None	This toggle is not used (“none”)
	TxPower=102,7	Display the TX Output power level
	quickpage1=Telemetry monitor	Long pressing the page up/down buttons cycle through the quick pages in forward/reverse order.

Micro Quad with throttle -100 to 100 display, bargraphs, timer and telemetry (on-board battery voltage)



[gui-128x64]		
	V-trim=0,11,1	We don't need to show the throttle trim
	H-trim=5,60,3	This displays the aileron trim
	V-trim=124,11,2	Displays the elevator trim
	H-trim=74,60,4	Displays the rudder trim
	Big-box=4,12,Ch9	Display throttle as percentage: Ch9 is set to match the output of Ch3 and scale it to 50% (a range of -50 to 50) which is then offset 50 to give a range of 0-100
	Small-box=4,28,Timer1	Countdown timer with sticky hold
	Small-box=4,38,Timer2	Stopwatch timer with sticky hold
	Small-box=75,48,Timer4	Permanent Timer
	Small-box=4,48,TelemV1	Display the telemetry value V1
	Toggle=88,8,0,72,0,HOLD	Displays when the throttle hold is engaged
	Toggle=60,8,228,229,229,SW A	Displays state of the LEDs
	Toggle=60,20,225,218,217,SW B	Displays flight mode state
	Toggle=60,35,212,217,0,FMODE	Displays flip mode
	Bargraph=75,23,Ch1	Bargraph the output for this channel
	Bargraph=81,23,Ch2	Bargraph the output for this channel
	Bargraph=87,23,Ch3	Bargraph the output for this channel
	Bargraph=93,23,Ch4	Bargraph the output for this channel
	Bargraph=99,23,Ch5	Bargraph the output for this channel
	Bargraph=105,23,Ch6	Bargraph the output for this channel
	Bargraph=111,23,Ch7	Bargraph the output for this channel
	Bargraph=117,23,Ch8	Bargraph the output for this channel
	TxPower=75,1	Display the TX Output power level
	Battery=102,1	Display the TX battery voltage

Micro Quad with throttle 0-100% display, light, flip and camera controls. Includes bargraph for channels 1-8, does not display model icon



[gui-128x64]		
	V-trim=0,11,0	We don't need to show the throttle trim
	H-trim=5,60,3	This displays the aileron trim
	V-trim=124,11,2	Displays the elevator trim
	H-trim=74,60,4	Displays the rudder trim
	Big-box=4,18,Ch9	Display throttle as percentage: Ch9 is set to match the output of Ch3 and scale it to 50% (a range of -50 to 50) which is then offset 50 to give a range of 0-100
	Small-box=4,38,Timer1	Countdown timer with sticky hold
	Small-box=4,48,Timer2	Stopwatch timer with sticky hold
	Small-box=75,48,Timer4	Permanent Timer
	Toggle=88,8,0,72,0,HOLD	Displays when the throttle hold is engaged
	Toggle=60,8,228,229,229,SW A	Displays state of the LEDs
	Toggle=60,20,1,193,194,SW B	Displays flight mode state
	Toggle=60,35,212,217,0,FMODE	Displays flip mode
	Toggle=60,47,0,208,0,Ch10	Displays camera status
	Bargraph=75,23,Ch1	Bargraph the output for this channel
	Bargraph=81,23,Ch2	Bargraph the output for this channel
	Bargraph=87,23,Ch3	Bargraph the output for this channel
	Bargraph=93,23,Ch4	Bargraph the output for this channel
	Bargraph=99,23,Ch5	Bargraph the output for this channel
	Bargraph=105,23,Ch6	Bargraph the output for this channel
	Bargraph=111,23,Ch7	Bargraph the output for this channel
	Bargraph=117,23,Ch8	Bargraph the output for this channel
	Battery=102,1	Display the TX battery voltage
	TxPower=75,1	Display the TX Output power level
	quickpage1=Model setup	Long pressing the page up/down buttons cycle through the quick pages in forward/reverse order.
	quickpage2=Channel monitor	

Tricks

A few tricks you can use on every model:

Name your virtual channels so you can tell what they do in the GUI:


```
[virtchan1]
name=video
template=complex
[mixer]
src=AIL
dest=Virt1
scalar=0
curvetype=fixed

[virtchan2]
name=photo
template=complex
[mixer]
src=AIL
dest=Virt2
scalar=0
curvetype=fixed

[virtchan3]
name=St-THold
template=complex
…
```

Sticky Throttle Hold:

Sticky throttle hold allows you to use a switch to drop the throttle when you want the motors to stop (e.g. right before a crash!). You can assign a single switch if desired. However, this version ensures that after you engage throttle hold you must also return the throttle to the lowest position before you can restart the motors.

First set the safety switch for the throttle channel

```
[channel3]
safetysw=Virt3
safetyval=-100
failsafe=-100
template=simple
[mixer]
src=THR
dest=Ch3
curvetype=5point
points=-100,-20,30,60,100
```

Then set up a virtual channel as a sticky hold

```
[virtchan3]
name=St-THold
template=complex
[mixer]
src=Virt3
dest=Virt3
usetrim=0
curvetype=min/max
points=0
[mixer]
src=Ch3
dest=Virt3
offset=-1
usetrim=0
muxtype=add
[mixer]
src=AIL
dest=Virt3
switch=HOLD1
usetrim=0
curvetype=fixed
```

Show Throttle 0-100% rather than -100 to 100:

By default, the user interface shows a throttle value of -100 (lowest position) to 100 (highest). But it may be more desirable to show the throttle as a percentage. Keep in mind:

- You must use a channel that the display can show. For Deviation release 4.0.1 the box will only show "real" channels so you can't use a virtual channel for this. More recent nightlies may allow you to use a virtual channel.
- A "real" channel allows you to set a safetysw so that the percentage display returns to 0 when you engage the throttle hold. You can also set failsafe, min, max and so on. Virtual channels cannot be used this way.
- The GUI won't allow you to edit more real channels than you've defined in your num_channels= setting. So in a 4-channel device, if you want to edit a higher real-channel, set num_channels to the higher number (e.g. 9), edit the results, then set the num_channels back to 4.

```
[channel9]
safetysw=Virt3
max=100
failsafe=-100
min=0
template=simple
[mixer]
src=Ch3
dest=Ch9
scalar=50
offset=50
```

Example Model.ini files

Model1.ini for Simple 4-channel Heli

```
name=4-Channel
mixermode=Advanced
icon=heli.bmp
[radio]
protocol=DEVO
num_channels=8
fixed_id=444
tx_power=150mW

[protocol_opts]
Telemetry=On

[channel1]
template=expo_dr
[mixer]
src=ELE
dest=Ch1
[mixer]
src=ELE
dest=Ch1
switch=FMODE1
scalar=60

[channel2]
safetysw=HOLD1
safetyval=-100
template=simple
[mixer]
src=AIL
dest=Ch2
[mixer]
src=AIL
dest=Ch2
switch=FMODE1
scalar=60

[channel3]
template=expo_dr
[mixer]
src=THR
dest=Ch3

[channel4]
template=expo_dr
[mixer]
src=RUD
dest=Ch4
[mixer]
src=RUD
dest=Ch4
switch=FMODE1
scalar=60

[trim1]
src=LEFT_V
pos=TRIMLV+
neg=TRIMLV-
[trim2]
src=RIGHT_V
pos=TRIMRV+
neg=TRIMRV-
[trim3]
src=LEFT_H
pos=TRIMLH+
```

```
[trim4]
neg=TRIMLH-
value=-1,0,0
src=RIGHT_H
pos=TRIMRH+
neg=TRIMRH-
[timer1]
type=countdown
src=Ch3
time=210
[timer2]
src=Ch3
[timer4]
type=permanent
src=Ch3
val=4058028
[safety]
Auto=min
[gui-128x64]
V-trim=59,10,1
H-trim=5,59,3
V-trim=65,10,2
H-trim=74,59,4
Small-box=2,22,Ch3
Small-box=2,31,Timer1
Small-box=2,40,Timer2
Model=75,20
Battery=102,1
Toggle=4,10,0,3,0,HOLD
Toggle=13,10,0,5,0,FMODE
Toggle=22,10,0,4,0,None
Toggle=31,10,0,0,0,None
Toggle=40,10,0,0,0,None
TxPower=102,7
```

Model3.ini for Proto-X (Hubsan X11)

```
name=ProtoXB
mixermode=Advanced
icon=PROTOX01.BMP
[radio]
protocol=Hubsan4
num_channels=7
fixed_id=444
tx_power=30mW

[protocol_opts]
vTX MHz=5885
Telemetry=On

[channel1]
template=expo_dr
[mixer]
src=AIL
dest=Ch1
scalar=60
curvetype=expo
points=-10,-10
[mixer]
src=AIL
dest=Ch1
switch=SW B1
scalar=80
curvetype=expo
points=-10,-10
[mixer]
src=AIL
dest=Ch1
switch=SW B2
scalar=110
```

```
curvetype=expo
points=-10,-10

[channel2]
template=expo_dr
[mixer]
src=ELE
dest=Ch2
scalar=60
curvetype=expo
points=-10,-10
[mixer]
src=ELE
dest=Ch2
switch=SW B1
scalar=80
curvetype=expo
points=-10,-10
[mixer]
src=ELE
dest=Ch2
switch=SW B2
scalar=110
curvetype=expo
points=-10,-10
```

```
[channel3]
safetysw=Virt1
safetyval=-100
template=simple
[mixer]
src=THR
dest=Ch3
curvetype=3point
points=-100,60,100
smooth=1

[channel4]
template=expo_dr
[mixer]
src=RUD
dest=Ch4
scalar=60
curvetype=expo
points=30,30
[mixer]
src=RUD
dest=Ch4
switch=SW B1
scalar=75
curvetype=expo
points=30,30
[mixer]
src=RUD
dest=Ch4
switch=SW B2
curvetype=expo
points=30,30
```

```
[channel5]
reverse=1
template=simple
[mixer]
src=SW A0
dest=Ch5

[channel6]
template=simple
[mixer]
src=FMODE1
dest=Ch6
curvetype=min/max
points=0

[channel7]
template=simple
[mixer]
src=Virt2
dest=Ch7
curvetype=zero/max
points=0

[channel8]
template=simple
[mixer]
src=Virt3
dest=Ch8
curvetype=zero/max
points=0
```

[channel9]
safetysw=Virt1
max=100
failsafe=-100
min=0
template=simple
[mixer]
src=Ch3
dest=Ch9
scalar=50
offset=50
[virtchan1]
template=complex
[mixer]
src=Virt1
dest=Virt1
usetrim=0
curvetype=min/max
points=0
[mixer]
src=Ch3
dest=Virt1
offset=-1
usetrim=0
muxtype=add
[mixer]
src=AIL
dest=Virt1
switch=HOLD1
usetrim=0
curvetype=fixed

[trim1]
src=Virt2
pos=TRIMLV+
step=193
value=-100,0,0
[trim2]
src=RIGHT_V
pos=TRIMRV+
neg=TRIMRV-
step=10
[trim3]
src=LEFT_H
pos=TRIMLH+
neg=TRIMLH-
step=10
[trim4]
src=RIGHT_H
pos=TRIMRH+
neg=TRIMRH-
step=10
value=-1,0,0
[trim5]
src=Virt3
pos=TRIMLV-
step=193
value=-100,0,0

```
[timer1]
type=countdown
src=Ch3
time=210
[timer2]
src=Ch3
[timer4]
type=permanent
src=Ch3
val=3864300
[telemalarm1]
source=Volt1
above=1
value=32
[safety]
Auto=min
```

```
[gui-128x64]
V-trim=0,11,1
H-trim=5,60,3
V-trim=124,11,2
H-trim=74,60,4
Big-box=4,12,Ch9
Small-box=4,28,Timer1
Small-box=4,38,Timer2
Small-box=75,48,Timer4
Small-box=4,48,TelemV1
Toggle=88,8,0,72,0,HOLD
Toggle=60,8,228,229,229,SW A
Toggle=60,20,225,218,217,SW B
Toggle=60,35,212,217,0,FMODE
Bargraph=75,23,Ch1
Bargraph=81,23,Ch2
Bargraph=87,23,Ch3
Bargraph=93,23,Ch4
Bargraph=99,23,Ch5
Bargraph=105,23,Ch6
Bargraph=111,23,Ch7
Bargraph=117,23,Ch8
Battery=102,1
TxPower=75,1
quickpage1=Telemetry monitor
quickpage2=Channel monitor
```

Model7.ini for JXD-392 (uses v2x2 protocol)

```
name=JXD-392
mixermode=Advanced
icon=PROTOX01.BMP
[radio]
protocol=V202
num_channels=10
fixed_id=1717
tx_power=30mW

[protocol_opts]
Re-bind=No

[channel1]
reverse=1
template=expo_dr
[mixer]
src=AIL
dest=Ch1
scalar=60
curvetype=expo
points=-10,-10
[mixer]
src=AIL
dest=Ch1
switch=SW B1
scalar=80
curvetype=expo
points=-10,-10
[mixer]
src=AIL
dest=Ch1
switch=SW B2
curvetype=expo
points=-10,-10
```

```
[channel2]
template=expo_dr
[mixer]
src=ELE
dest=Ch2
scalar=60
curvetype=expo
points=-10,-10
[mixer]
src=ELE
dest=Ch2
switch=SW B1
scalar=80
curvetype=expo
points=-10,-10
[mixer]
src=ELE
dest=Ch2
switch=SW B2
curvetype=expo
points=-10,-10

[channel3]
safetysw=Virt3
safetyval=-100
failsafe=-100
template=simple
[mixer]
src=THR
dest=Ch3
curvetype=5point
points=-100,-20,30,60,100
```

[channel4]
reverse=1
template=expo_dr
[mixer]
src=RUD
dest=Ch4
scalar=80
curvetype=expo
points=0,0
[mixer]
src=RUD
dest=Ch4
switch=SW B1
curvetype=expo
points=0,0
[mixer]
src=RUD
dest=Ch4
switch=SW B2
scalar=125
curvetype=expo
points=0,0
[channel5]
reverse=1
template=simple
[mixer]
src=SW A0
dest=Ch5

[channel6]
template=simple
[mixer]
src=FMODE1
dest=Ch6
curvetype=expo
points=0,0
[channel7]
template=simple
[mixer]
src=Virt1
dest=Ch7
curvetype=zero/max
points=0
[channel8]
max=100
min=-100
template=simple
[mixer]
src=Virt2
dest=Ch8
curvetype=zero/max
points=0

```
[channel9]
safetysw=Virt3
max=100
failsafe=-100
min=0
template=simple
[mixer]
src=Ch3
dest=Ch9
scalar=50
offset=50

[channel10]
template=simple
[mixer]
src=Virt4
dest=Ch10

[virtchan1]
template=complex
[mixer]
src=AIL
dest=Virt1
scalar=0
curvetype=fixed

[virtchan2]
template=complex
[mixer]
src=AIL
dest=Virt2
scalar=0
curvetype=fixed
```

```
[virtchan3]
template=complex
[mixer]
src=Virt3
dest=Virt3
usetrim=0
curvetype=min/max
points=0
[mixer]
src=Ch3
dest=Virt3
offset=-1
usetrim=0
muxtype=add
[mixer]
src=AIL
dest=Virt3
switch=HOLD1
usetrim=0
curvetype=fixed

[virtchan4]
template=simple
[mixer]
src=Virt2
dest=Virt4
curvetype=fixed

[trim1]
src=Virt1
pos=TRIMLV+
step=193
value=-100,0,0
```

[trim2]
src=RIGHT_V
pos=TRIMRV+
neg=TRIMRV-
step=5
[trim3]
src=LEFT_H
pos=TRIMLH+
neg=TRIMLH-
step=10
[trim4]
src=RIGHT_H
pos=TRIMRH+
neg=TRIMRH-
step=5
[trim5]
src=Virt2
pos=TRIMLV-
step=193
value=-100,0,0
[trim6]
src=Virt4
pos=TRIMLV+
step=192
value=-100,0,0
[timer1]
type=countdown
src=Ch3
time=330
[timer2]
src=Ch3
[timer4]
type=permanent
src=Ch3
val=9670136

[safety]
Auto=min
[gui-128x64]
V-trim=0,11,0
H-trim=5,60,3
V-trim=124,11,2
H-trim=74,60,4
Big-box=4,18,Ch9
Small-box=4,38,Timer1
Small-box=4,48,Timer2
Small-box=75,48,Timer4
Toggle=88,8,0,72,0,HOLD
Toggle=60,8,228,229,229,SW A
Toggle=60,20,1,193,194,SW B
Toggle=60,35,212,217,0,FMODE
Toggle=60,47,0,208,0,Ch10
Bargraph=75,23,Ch1
Bargraph=81,23,Ch2
Bargraph=87,23,Ch3
Bargraph=93,23,Ch4
Bargraph=99,23,Ch5
Bargraph=105,23,Ch6
Bargraph=111,23,Ch7
Bargraph=117,23,Ch8
Battery=102,1
TxPower=75,1
quickpage1=Model setup
quickpage2=Channel monitor

Legal Notice about Deviation Software

Disclaimer

Deviation is experimental software. There are no guarantees made or implied about the quality or reliability of this software. RC models can cause serious injury or even death if flown improperly. By deciding to use the Deviation software, you are taking sole responsibility for the control of your models. The authors of Deviation will not be held responsible for any injury or damage whatsoever caused by the use of the Deviation firmware. Be careful and cautious.

Notices

Deviation is an independent work. The Deviation project is not affiliated, supported or acknowledged by Walkera®. The authors have never been in contact with Walkera nor do they know of Walkera's stance on this project. The Deviation team provides no guarantee that the Deviation firmware will not harm your transmitter (although this should not be possible). There is also no guarantee that Walkera® will not make changes to future versions of the hardware, firmware or Dfuse tool which would make it incompatible with Deviation.

Legal status and copyright

The Deviation Project is hosted at www.deviationtx.com and the Source are available at http://bitbucket.org/deviationtx/deviation.