A second review of FrSky OpenTx equipment

I have started to replace the default transmitter voice messages with my own using Audacity to do the recording. Its very straightforward to copy files into the appropriate folder (directory) on the SD memory card provided you have a card reader device on your computer. Its a good idea to back up the card while you are at it, in case of a disaster wiping it. I like the fact that all model and setup data is held on the card but it does make it vulnerable if you forget to backup. So don't. I will make a copy on another card to take to the field, just in case. Low speed MicroSD cards are very cheap. I think that the data is transferred once when a model is selected so the card probably does not need to be a fast one.

The balance bar is an oddity. I didn't see the point of it till I field-tested the Tx when hand-launching a glider without using a transmitter tray. If you clip a neckstrap into the hole on the Tx it tilts strongly downwards and is very awkward to hold. I suspect the original design had a lighter battery. The balance bar allows you to shift the neckstrap clip to a hole so the balance feels right.

To start with I used a curve for the aileron and elevator servos. You build a curve by moving up to 17 dots up or down in a four quadrant graph area. In the one curve you can set both expo and throw. I set mine to have low movement near neutral and an increasing sensitivity as the stick is moved. However this also affected the trim so I got very little trim movement. I tried an option that allows you to continue to full throw on trim but this seemed a bit of a kludge. I then tried setting expo instead of curve and it turned out that this was better for trim.

One of the confusing things in OpenTx is the fact that it uses words that are different from the usual ones. The first time I came across this was setting servo rates. I realised that in OpenTx this is called 'weight'. A weight of 100 gives full throw and the amount goes down in proportion to this number, which is in effect a percentage. Once you grasp this, setting dual or even triple rates becomes easy. In the INPUTS screen you add new lines of code each of which corresponds with a weight (rate) and a switch setting. Using a three-position switch allows three different rates. In the same line you can set the degree of expo curve. You can also set the speed at which the settings change when you move the switch.

Here is some sample code:

```
Ail        60    Ail    E40    SF↑
           100   Ail           SF↓
```

The first line gives a weight (rate) of 60% and expo of 40 when switch F is up.
The second line gives a weight (rate) of 100% and no expo when switch F is down.
Moving the switch while you program puts the selected line in bold to allow you to check the settings for each position.
You could set up a rotary proportional control, of which there are four on the Taranis, to make this even more variable. Perhaps a new line for each 10% of rotation.

There is a separate OUTPUTS screen for setting maximum servo throw and direction. Caution is needed as if you set weight to 50 in Inputs and then again to 50 in Outputs, they each have an effect, so you get 25. You can also set signal timing on this screen if your servos or receiver do not do this automatically. So a combination of weight, expo and throw allows very flexible setting and there are curves as well if you really need them.

In between the INPUTS and OUTPUTS screens there is the MIXER screen. Here you link inputs with outputs. As usual you can mix more than one input. In OpenTx this is very flexible. Each input can be given a weight in the mix. For example if you have flaps and a servo for each aileron you can mix them so that a switch or rotary control pushes both flaps down and both ailerons up to form a powerful airbrake but can mix in some aileron stick input to allow roll to level up for landing.

One great strength of the FrSky kit is telemetry. It works well and is astonishingly cheap. FrSky has now settled on a simple receiver connection method called SmartPort, though it also has a connection for the S.Bus system used by other makers. SmartPort is an extra servo-style connection on the receiver. Each sensor has two sets of three pins onto which the other end of the servo-style leads are plugged. You connect the first sensor to the receiver, then use the second set of sensor pins to daisy chain the second sensor. That can then daisy chain the third and so on. There is no need for a telemetry controller. Using minimum-length leads you can create a very neat and light set-up. The Rx usually detects the sensors automatically.

The telemetry sensors are very light but rather fragile, especially the battery voltage one that I used first. I pulled one of these apart and broke it when removing it from the velcro mount. I used transparent heat shrink to hold the next one together. There is a display screen on the voltage sensor that gives the voltage in each cell but this does not seem very useful if you do balanced charging, so it is best to put the voltage sensor out of the way and use a JST-XH extension lead. Then you do not need to keep plugging and unplugging the battery balance lead into the sensor itself. The sensor has pins to take up a six-cell battery and you use two sensors in series to handle up to twelve-cell batteries.

I then tried the GPS sensor. This also worked faultlessly and enabled display of height, position, speed and so on. To use the vario sensor you set up switches and functions to detect the data and output the results. There is a sound function, called Vario, built into the Transmitter that will produce the usual vario tones. You can also set up a periodic voice message telling you your height or when you are at the legal maximum height. You can, of course, do this using the GPS sensor instead. The sounds come through the Tx loudspeaker, or into earpieces if you prefer not to annoy people. There is another sensor for RPM, using a takeoff from two of the ESC outputs, that also has two temperature sensors for motor, ESC or battery. Others allow cell current and airspeed to be monitored.

Learning to set up telemetry screens is a bit awkward the first time, but when you have done it once it really is very simple. You choose what data to display and whether the display is in digital or bar format. One long press on a button brings the screen up before, or as, you fly.

One feature I found useful was to make a copy of a model before I started editing it. If I made a complete mess I could delete it and revert to the copy. This is of course standard practice in computer coding. Programmers are cautious by nature.

Peter Scott
31 May 2016