SilF4ware

- RCGroups thread
- Github Repository

Receiver

NRF24

The mini module is the lightest option and is recommended for micro quads. The module with PA+LNA (GT-24) provides a bigger range, but weight and power consumption is higher.

Mini

- Weight: 0.49g
- It can be found using “nrf24l01 mini” as search term
- Pinout
- where to cut PCB for adding a wire antenna: rcgroups

PA LNA (GT-24)

- Weight: 1.08g
- It can be found with “GT-24” as search term on Banggood (and much cheaper on AliExpress)
- It is possible to use a U.FL antenna when moving one component, see the picture in this post for instructions.

XN297

- Can be harvested from toy transmitters like H101, H8, …
- DIY PCB
- There is a XN297L module commercially available (Banggood), but no one has tested it yet.

Flight Controller

Basic Configuration

Main configuration is done in SilF4ware/config.h.

Receiver

The default configuration is setup for NRF24 modules. If using a NRF24 module with PA LNA, it is recommended to adjust TX_POWER in SilF4ware/config.h:

```
#define TX_POWER 1 // 0 .. 3 (use 1 when using an nRF24L01 PA LNA module)
```

If using a XN297 module, see radio_config.txt file for configuration notes.

Battery Cell Count

Default setup is configured for 4S batteries. Make sure to adjust CELL_COUNT_UNSCALED in SilF4ware/battery.c if needed. As an example, for a 2S setup:

```
#define CELL_COUNT_UNSCALED 2 // Voltage divider, idle_offset, and PID
values tuned for 4S.
```

Dshot

Default setup is configured for using Dshot 300+RPM Filter. If using RPM Filter, the number of magnets on the motor bell needs to be configured correctly in SilF4ware/drv_dshot_bidir.c:

```
#define MOTOR_POLE_COUNT 14 // usually on 22xx motors and above
// #define MOTOR_POLE_COUNT 12 // usually on 18xx motors and below
```

If using conventional D-Shot, adjust SilF4ware/hardware.h:

```
#define DSHOT_DMA_BIDIR // needed for RPM_FILTER, 4k loop frequency max
// #define DSHOT_DMA_DRIVER // conventional Dshot, consumes less cycles,
works for 8k loop frequency
// #define DSHOT_DRIVER // delay version
```

2D/3D Flying

3D flying is enabled by default. If using a 2D setup, following changes are needed:

In the main config file (config.h):

```
//#define INVERTED_ENABLE
#define FN_INVERTED CH_OFF
//#define LEVEL_MODE_INVERTED_ENABLE // be careful when enabling this
```

In the dshot configuration file (when using RPM Filter: drv_dshot_bidir.c):

```
// Enable this for 3D. The 'Motor Direction' setting in BLHeliSuite must be
set to 'Bidirectional' (or 'Bidirectional Rev.') accordingly:
//#define BIDIRECTIONAL
```

Misc

- Props out configuration is enabled by default (comment INVERT_YAW_PID to disable it)
- Default rates are very high, adjust them if needed
- PID configuration for acro mode is done in SilF4ware/pid.c and for level mode in SilF4ware/angle_pid.c_PID to disable it)

Analog Aux Channels

Quoting from this post:

The variables aux_analog[0] and aux_analog[1] hold a value between 0.0 and 2.0 which you can use in various places in the code. Per default they are used to tweak Kp and Kd respectively. If you want to use them for something else, change the define for AA_pidkp and AA_pidkd to look similar to the one for AA_pidki, i.e. #define AA_pidkp 1.0f and so on.

Now you could use aux_analog[0] and aux_analog[1] for example to tune the filter frequency by adding it to config.h like this:

```
#define GYRO_LPF_2ND_HZ_BASE 400 * aux_analog[ 0 ]
#define GYRO_LPF_2ND_HZ_MAX 400 * aux_analog[ 1 ]
```

The possibilities are endless :)

Using

Gestures

- PID Tuning: some gestures have been swapped in comparison to other silverware branches: UDD switches to the next column and UDU to the next row
- LRU: reboot flight controller (nice when otherwise one would unplug and replug the battery)
- LRD: switch to DFU mode (nice if the BOOT button is mechanically hard to reach after installing the FC into a quad)

Motors test mode

SilF4ware has an improved MOTORS_TO_THROTTLE mode. It is not neccessary anymore to change IDLE_OFFSET to make sure that only one motor spins.

> With the LLU stick gesture you can try which motor spins up if you push the stick in the corresponding direction. LLD turns the MOTORS_TO_THROTTLE_MODE off again.

Devo TX

A Devo 7E build and model file which is able to display PID values can be found in this post. Note that it will only display PIDs set via gestures, not when set via analog aux channels.

Blackbox Logging

Blackbox logging is possible with an external logging device. See details here