Lighthouse implementation

Goal

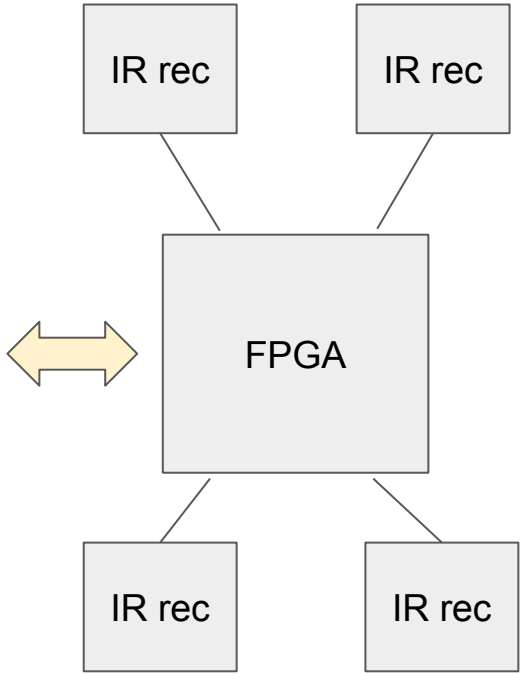
An understanding of the hardware design, where the algorithms are implemented and where to find the code

Parts

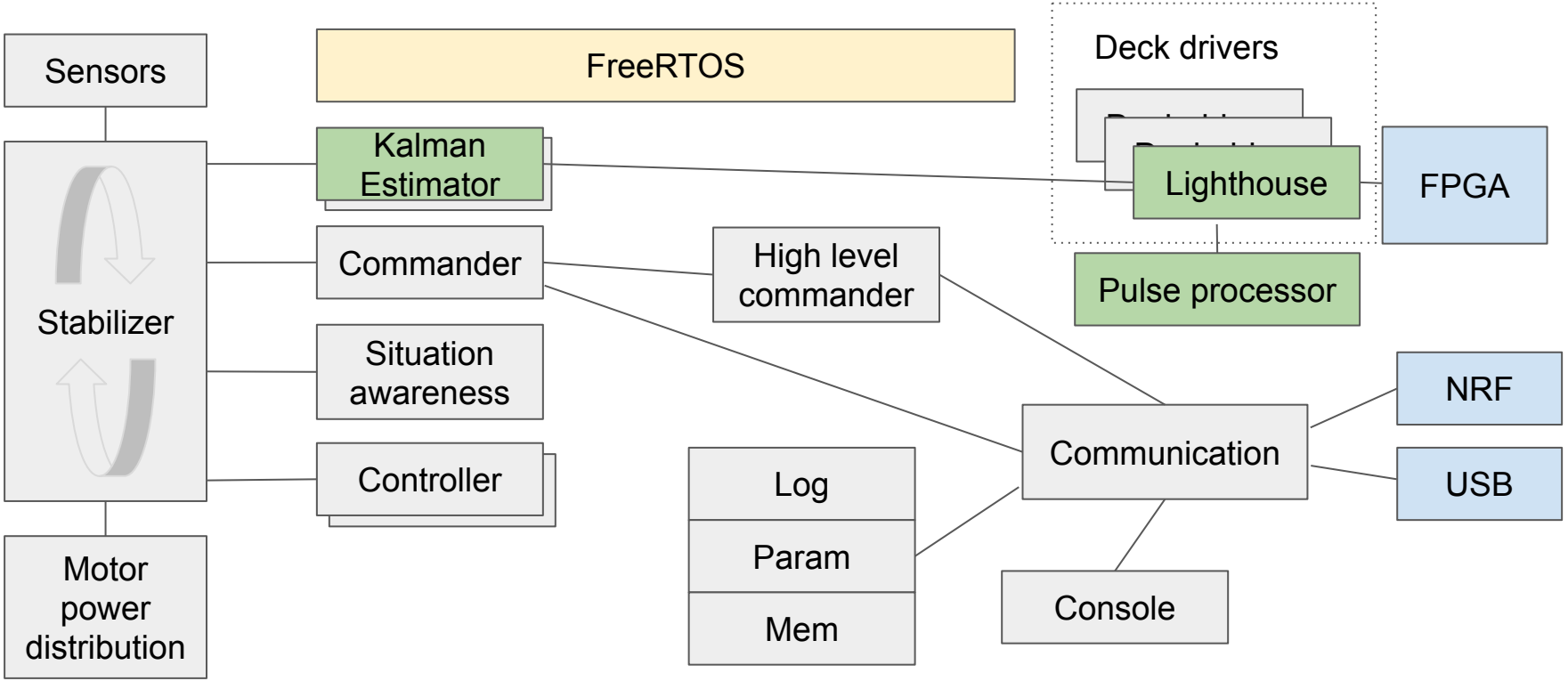
- The Lighthouse deck for the Crazyflie
- Lighthouse base stations

The Lighthouse deck

- 4 TS4231 IR receivers
- ICE40UP5K FPGA for signal processing
  - Detects pulses and frames
  - Timers to measure event times



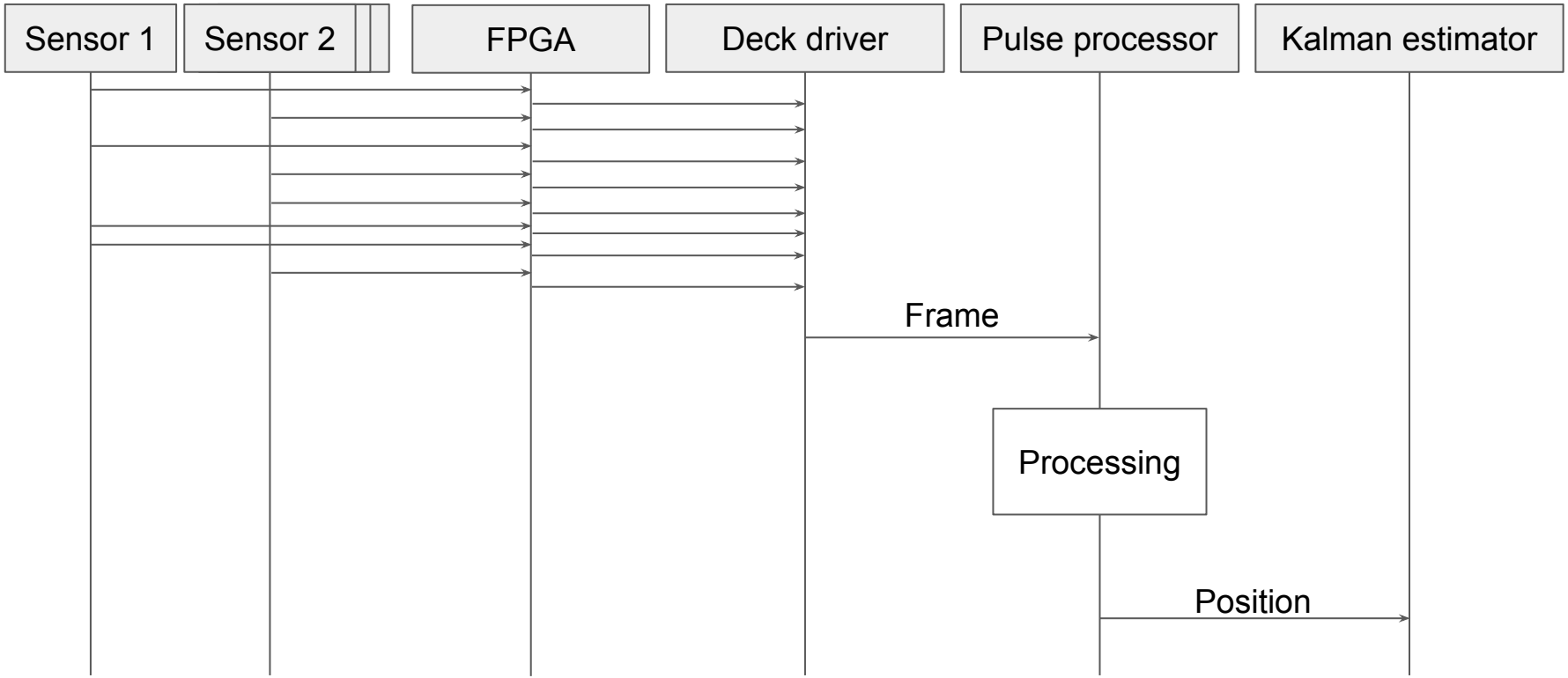
Crazyflie firmware overview with Lighthouse deck



Current Crazyflie firmware Lighthouse deck driver

- Runs in separate task
- Uses the kalman estimator (automatically enabled)
- Position is calculated in the deck driver by calculating crossing of vectors from two base stations
- Position is fed to the Kalman filter
- Decodes base station calibration data
- Memory mapping for base station geometry data
- Supports Lighthouse V1
- github.com/bitcraze/crazyflie-firmware/blob/2019.09/src/deck/drivers/src/lighthouse.c

Crazyflie firmware, Lighthouse flow chart



Future Crazyflie firmware Lighthouse deck driver

- Base station vectors fed into the kalman filter
- Position calculated in the kalman filter
- Will support positioning from one base station

Python lib Lighthouse support

- Memory mapping for base station geometry data

Future additions

- Estimation of base station geometry data using the Lighthouse deck
- Client support
- One base station
- Lighthouse V2

Conclusions

- Lighthouse V1
- Two base stations