Protocol specification

- protocol desing
  - 1st byte – type of message
  - 2nd byte – priority of message (for now for messages with higher priority than 0 new thread will be created)
  - 3rd and 4th byte – gives information about size of message – 3rd is 8 higher bytes, 4th is lower 8 bytes
  - 5 – value to be added to check sum – first 5 bytes need to be divisible by 7
  - ... – message
  - each message is terminated with following sequence – 0x00, 0x00, 0xFF, 0xFF, 0xFF
- ranges
  - 0x00 – 0x20 – settings management
  - 0x21 – 0x40 – control
  - 0x41 – 0xFF – telemetry
- * symbol means code is only used as get request/confirmation
- structures used are specified in `<protocol_spec.h>`

Settings

0x01	PI+CL	ping – empty message send by both sides
0x02	CL	restart of system
0x03	CL	shutdown
0x04	CL	disconnect client
0x08	CL	camera settings

Control

- these messages are only send by the client to raspberry pi
- standart control message will be send continuously – every few milliseconds (client will have counter to prevent overloading of the server)
- we are fine with few of these messages getting lost, as even minor movements will probably result in new message getting send
- special message will send confirmation to the client
  - client will add these messages to queue, which will be scanned from time to time
  - if response from server wasn't received in some time client will resend the message

0x21	CL+PI*	standart control message – status of analog joystick and shoulder buttons
0x22	CL+PI*	special control event – generated by pressing buttons that don't directly control control surfaces, but are free for further additions

Telemetry

- all telemetry, except pwm settings, is send in intervals
- if client want new data immediately it can send message with same header
- in such case using priority option is advisable
- range above 0x80 is used for errors

0x41	PI*+CL	status of all i/o
0x42	PI*+CL	general packet including all information
0x43	PI*+CL	attitude sensor and GPS
0x44	PI*+CL	battery status (reading from ina226)
0x45	PI*+CL	pwm setting on servo driver
0x81	PI	general error message