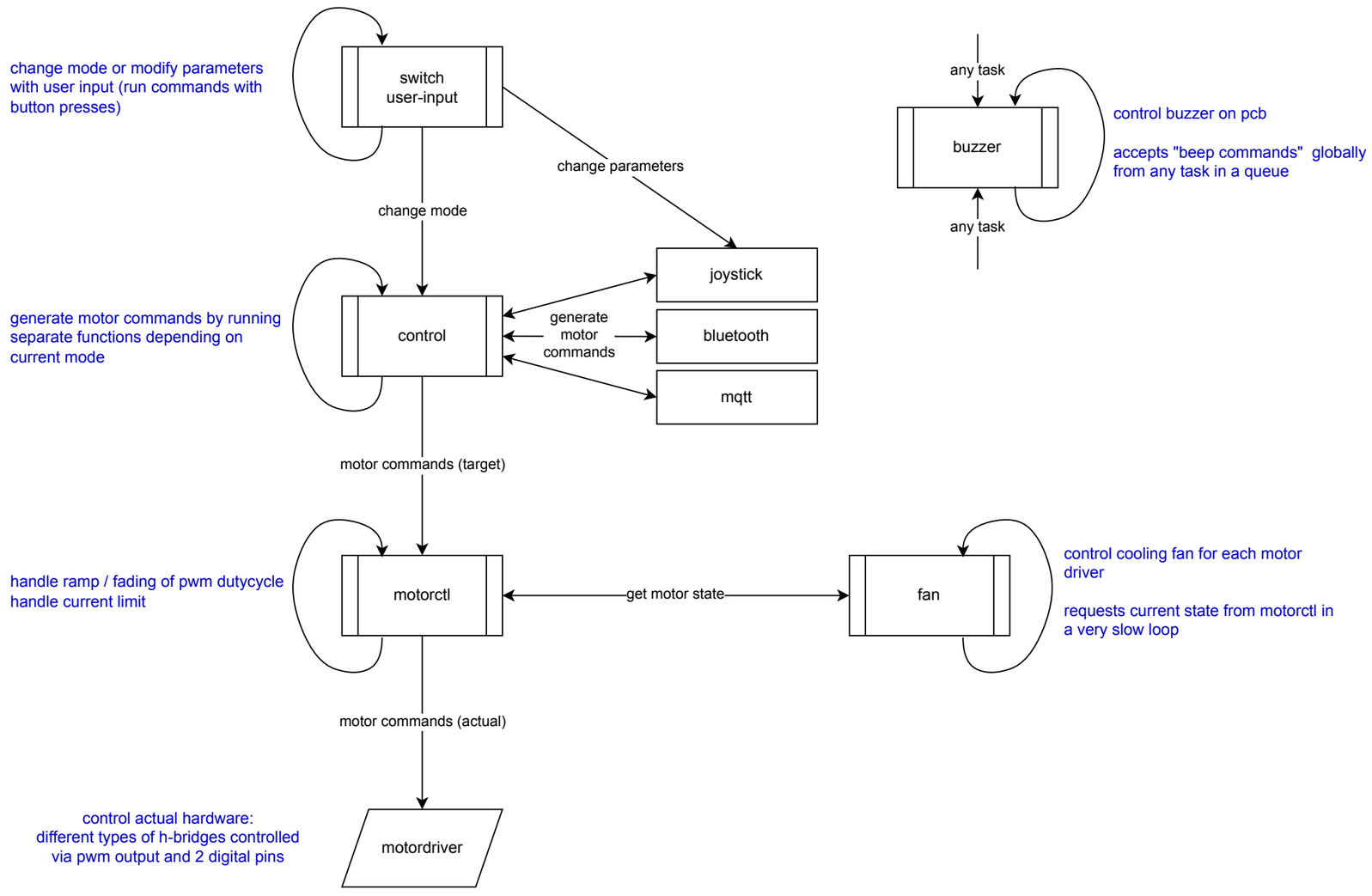
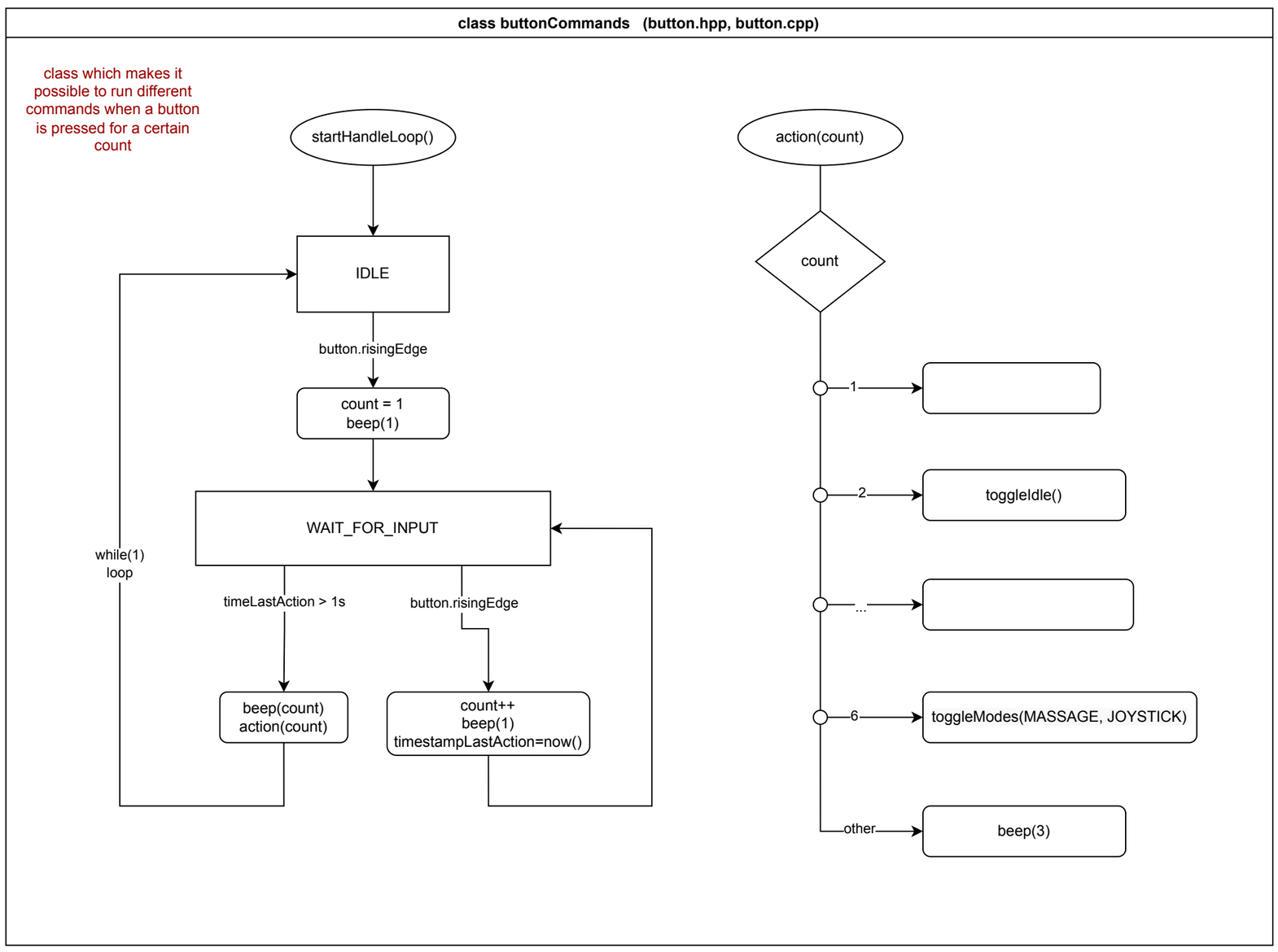
Tasks / Threads overview



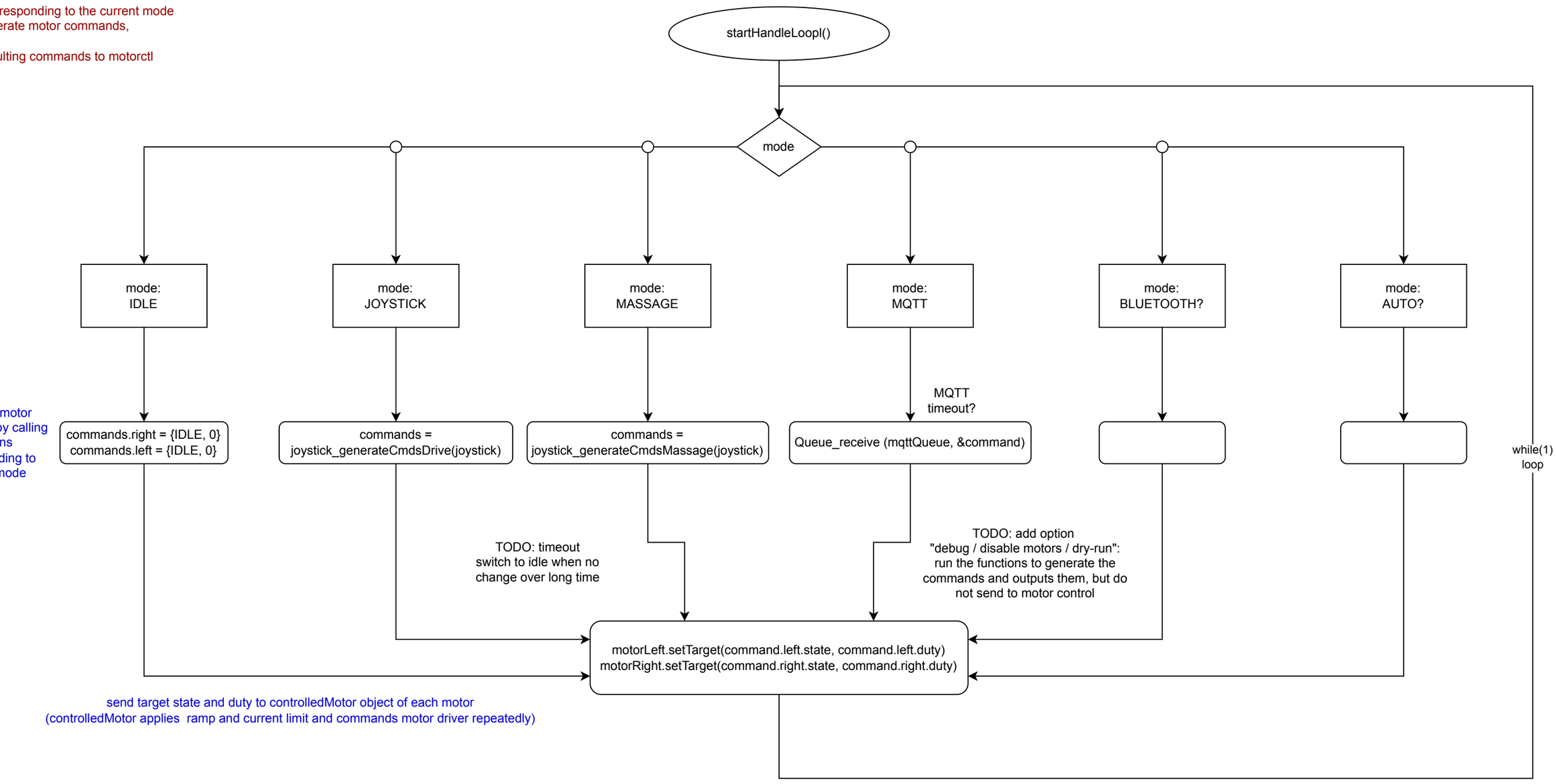
class buttonCommands (button.hpp, button.cpp)



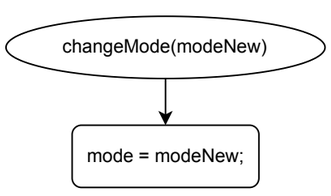
class controledArmchair (control.hpp, control.cpp)

run functions corresponding to the current mode that generate motor commands,

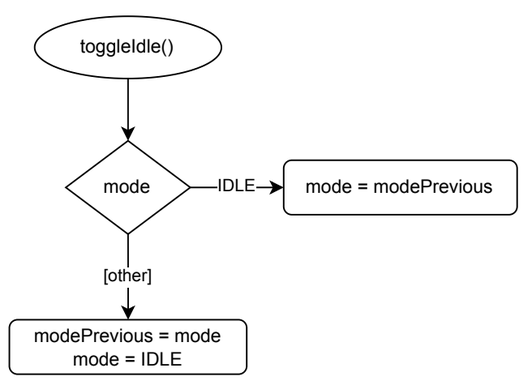
send the resulting commands to motorctl



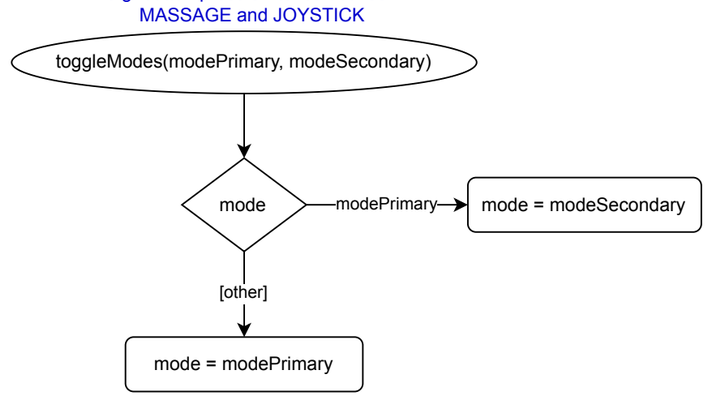
switch to specific mode



toggle between IDLE and previous or default mode



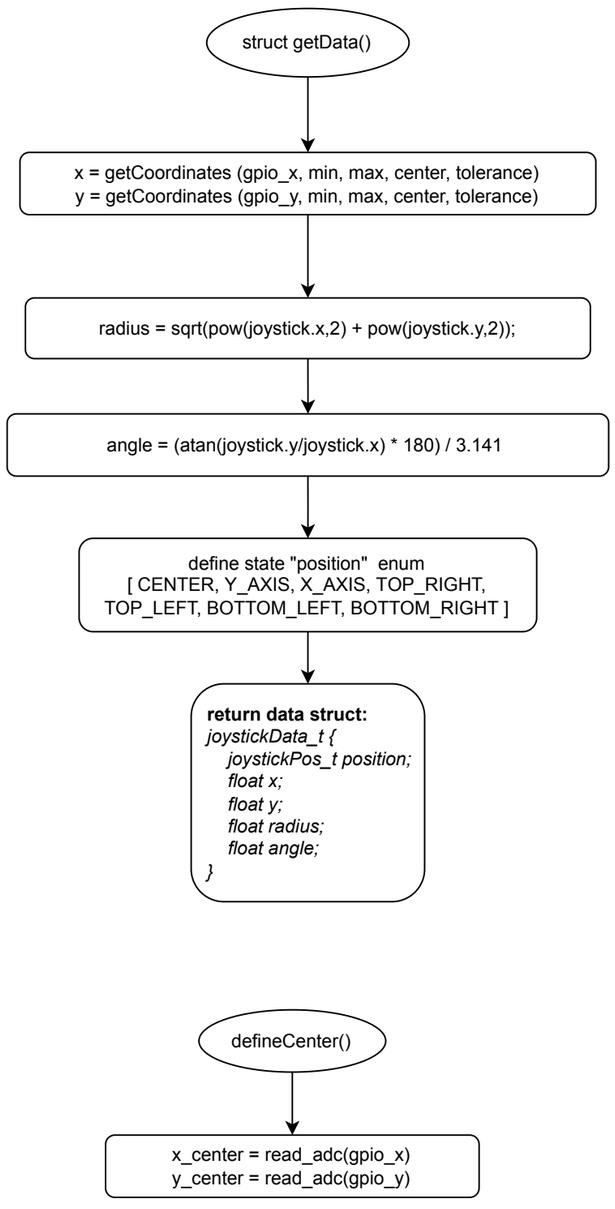
toggle between two specific modes
e.g. button press switches between MASSAGE and JOYSTICK



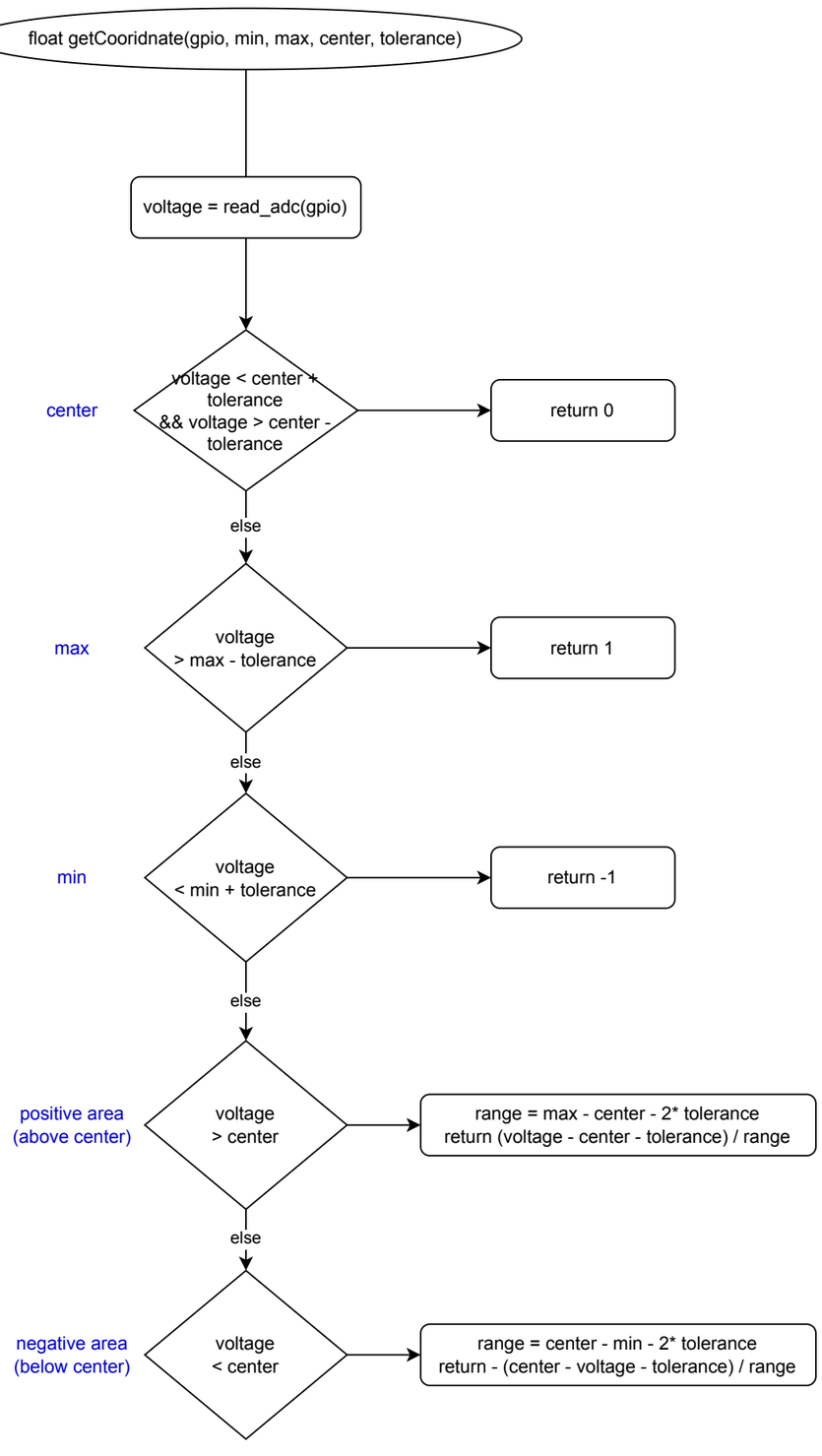
class evaluatedJoystick (joystick.hpp, joystick.cpp)

function that makes it possible to get a struct with current state and data of the joystick.

this can be used as input in other functions or tasks

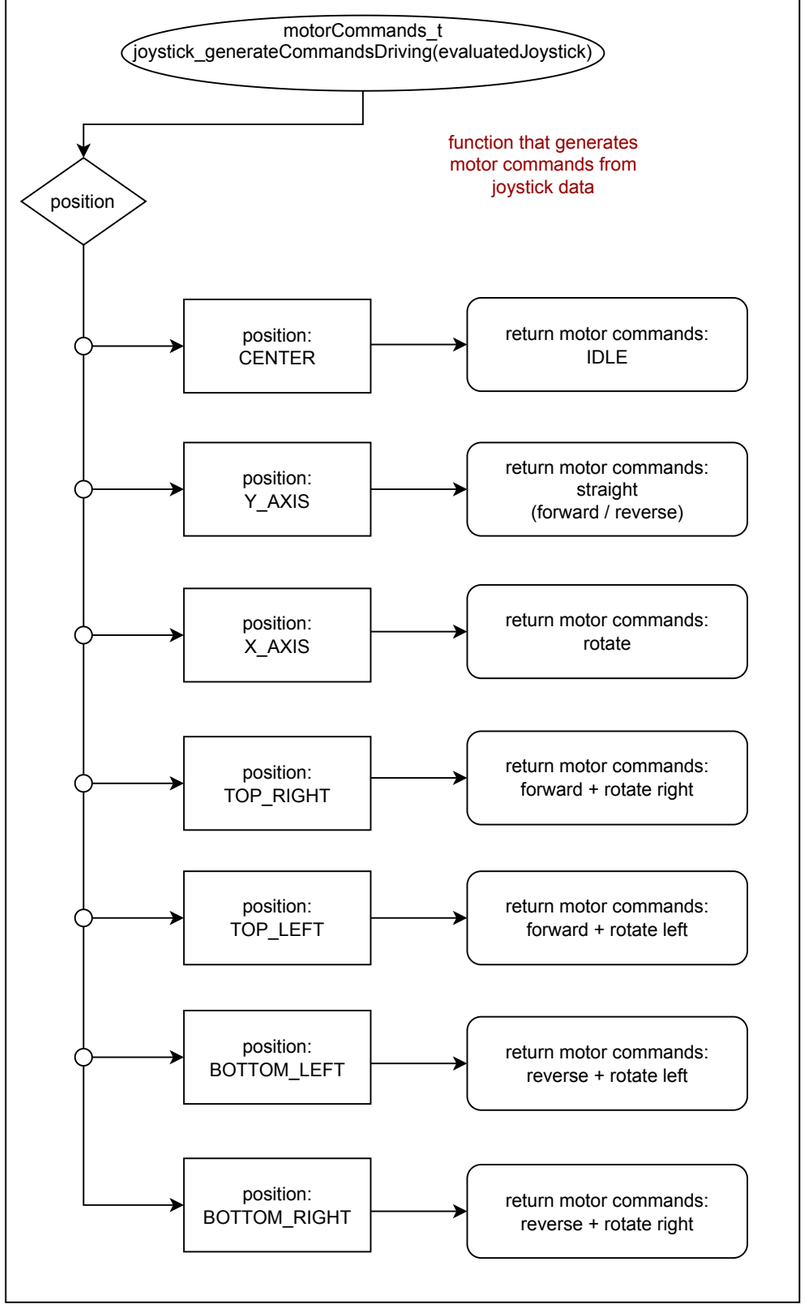


private function that reads an analog input and calculates coodrinates according to given parameters



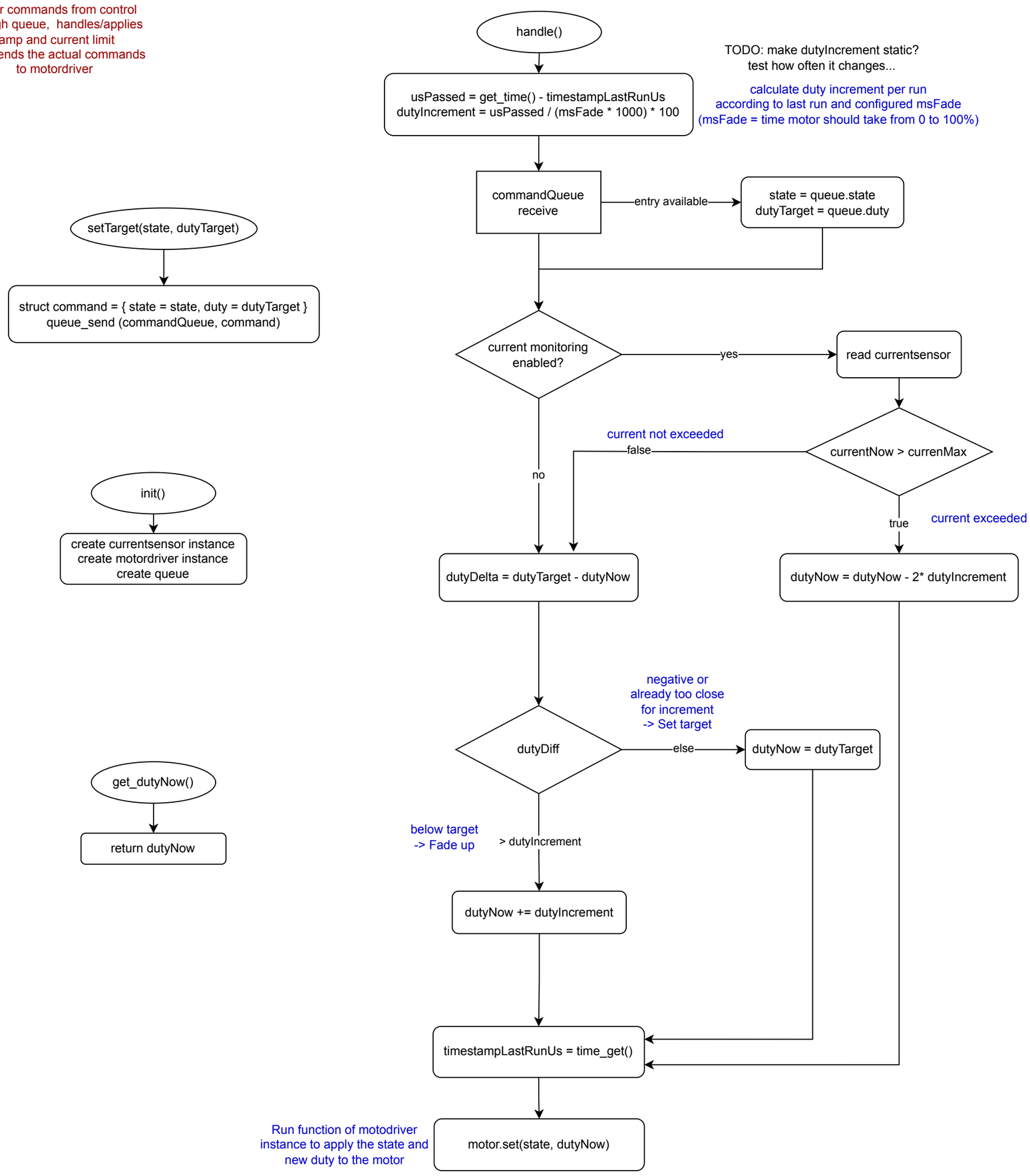
function: joystick_generateCommandsDriving

function that generates motor commands from joystick data

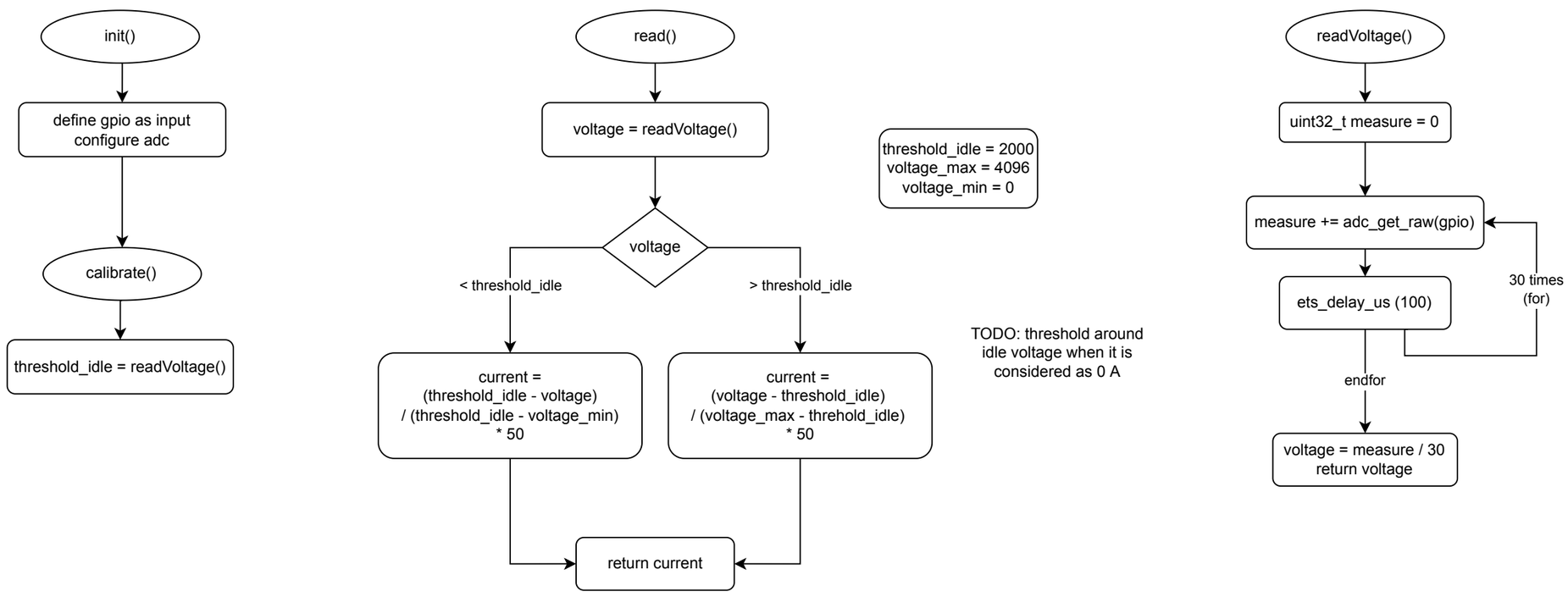


class controlledMotor (motorctl.hpp, motorctl.cpp)

class for each motor that receives motor commands from control through queue, handles/applies ramp and current limit then sends the actual commands to motordriver

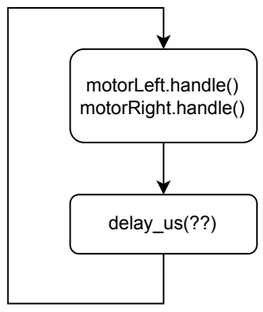


class currentsensor



DEL ### - old unneeded or temporary flowcharts

task motorCtl



global config

config.cpp only:
struct motorconfig_t config_motorLeft {... }
struct motorconfig_t config_motorRight {...}

config.hpp + config.cpp:
handledMotor **motorLeft** (& config_motorLeft);
handledMotor **motorRight** (& config_motorRight);