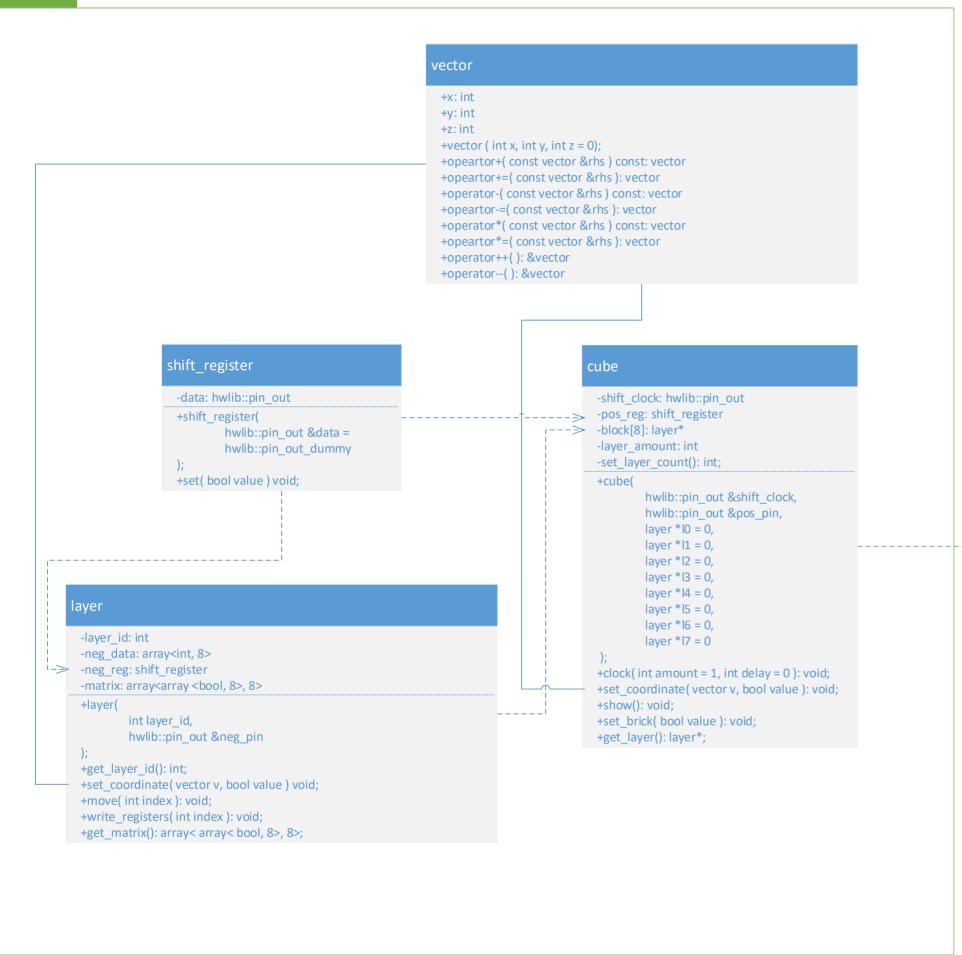
CUBE LIBRARY



CUBE API -----> #c: cube #frame_delay: int #frame_count: int +animation(const cube &c, int frame_delay = 5, int frame_count = 0 +draw(hwlib::pin_in &interrupt = hwlib::pin_in_dummy): virtual void = 0; +set_frame_delay(int delay): void; rectangle -start, end: vector -start, end: vector ---> -left, right, top, bottom: line -filled: bool const cube &c, vector start, +rectangle(vector end const cube &c, vector start, +draw(vector end, hwlib::pin_in &interrupt = bool filled = false hwlib::pin_in_dummy): void override; +draw(hwlib::pin_in &interrupt = hwlib::pin_in_dummy): void override; CUBE APPLICATION pulsing_cube alphabet rotating_plane +pulsing_cube(-current_char: char +rotating_plane(const cube &c, -current_char_data: array< array< bool, 8>, 8> const cube &c, int frame_delay int frame_delay +alphabet(const cube &c +draw(hwlib::pin_in &interrupt = hwlib::pin_in &interrupt = +set_char(hwlib::pin_in_dummy hwlib::pin_in_dummy char capital_letter,): void override;): void override; int offset): void; +offset(array< ::array< bool, 8>, 8> character, int amount): array< array< bool, 8>, 8> +draw(-mode: char hwlib::pin_in &interrupt = +kitt(hwlib::pin_in_dummy const cube &c,): void override; int frame_delay, char mode = '/' +draw(hwlib::pin_in &interrupt = hwlib::pin_in_dummy): void override;