

## stepmotor5.h

```
1 #ifndef INC_STEPMOTOR5_H_
2 #define INC_STEPMOTOR5_H_
3
4 #include "stm32f4xx_hal.h"
5
6 #define STEPS_PER_REV_MOTOR5 200 /* liczba krokow na pelen obrot silnika */
7 #define MICROSTEP_NUM_MOTOR5 32 /* liczba mikrokrokow -> 1 = brak
   mikro krokow */
8
9 #define PULSE_TIM_MOTOR5 htim2 /* timer odpowiedzialny za generowanie
   sygnalu PWM */
10 #define PULSE_TIM_CH_MOTOR5 TIM_CHANNEL_2 /* kanal generowania sygnalu PWM */
11
12 #define XT_TIM_CLK_MOTOR5 84000000 /* czestotliwosc wejsciuwa timera PULSE_TIM
   w Hz */
13 #define XT_TIM_PSC_MOTOR5 41 /* prescaler timera PULSE_TIM */
14
15 #define STEPPER_CW_MOTOR5 0 /* stan pinu DIR podczas ruchu CW */
16 #define STEPPER_CCW_MOTOR5 1 /* stan pinu DIR podczas ruchu CCW */
17
18 void stepper_init_motor5(void);
19 void stepper_speed_motor5(uint16_t rpm);
20 void stepper_run_motor5(uint8_t dir);
21 void stepper_stop_motor5(void);
22 void stepper_steps_motor5(uint16_t steps, volatile uint16_t *stepLimit5);
23 void stepper_rot_motor5(uint16_t ang, uint16_t rpm, uint8_t dir, volatile uint16_t
   *stepLimit5, volatile int* rotationCounter5, volatile uint16_t* isStop5);
24 void stepper_rot_home_motor5(uint16_t rpm, uint8_t dir, volatile uint16_t *stepLimit5,
   volatile int* rotationCounter5, volatile uint16_t* isStop5);
25 #endif /* INC_STEPMOTOR5_H_ */
26
```