```c
/* Name: David (DongYun) Kim
 * SID: 200405213
 * Course: ENEL351
 * Description: ENEL351 Project - Smart Parking System
 * File name: pwm.c
 */

#include "stm32f10x.h"
#include "pwm.h"

void pwminit(void) // Setup the clocks, IO pin, and timer config for PWM using Timer 3 CH1
{
 // Configure PA6 and PA7 as AFIO ( Write 1011b into the configuration and mode bits )
 RCC->APB2ENR |= RCC_APB2ENR_IOPAEN | RCC_APB2ENR_AFIOEN;
 GPIOA->CRL |= GPIO_CRL_CNF6_1 | GPIO_CRL_MODE6_1 | GPIO_CRL_MODE6_0;
 GPIOA->CRL &= ~GPIO_CRL_CNF6_0;

 GPIOA->CRL |= GPIO_CRL_CNF7_1 | GPIO_CRL_MODE7_1 | GPIO_CRL_MODE7_0;
 GPIOA->CRL &= ~GPIO_CRL_CNF7_0;

 // Configure TIM3 CH1 as PWM
 RCC->APB1ENR |= RCC_APB1ENR_TIM3EN; // Turn on the Timer 3 clock
 TIM3->CR1 = 0; // Disable Timer 3
 TIM3->CR1 |= TIM_CR1_ARPE; // Enable AutoReload

 // PWM mode 1
   TIM3->CCMR1 |= TIM_CCMR1_OC1M_2 | TIM_CCMR1_OC1M_1;
   TIM3->CCMR1 |= TIM_CCMR1_OC1PE | TIM_CCMR1_OC1FE; // Preload Enable, Fast Enable
   // PWM mode 2
   TIM3->CCMR1 |= TIM_CCMR1_OC2M_2 | TIM_CCMR1_OC2M_1;
   TIM3->CCMR1 |= TIM_CCMR1_OC2PE | TIM_CCMR1_OC2FE; // Preload Enable, Fast Enable

 TIM3->CCER |= TIM_CCER_CC1E; // Enable CH1
 TIM3->CCER |= TIM_CCER_CC2E; // Enable CH2


 TIM3->PSC = 720 - 1; // Divide 72 MHz by 720 (PSC+1), PSC_CLK= 100000 Hz, 1 count = 10 uS
 TIM3->ARR = 1000;
 TIM3->CCR1 = 500;
 TIM3->CCR2 = 500;

 TIM3->EGR |= TIM_EGR_UG; // Generate Timer Update Event
 TIM3->CR1 |= TIM_CR1_CEN; // Enable Timer3
}

/*** Servo Motor at the Entry Gate ***/
void change_CH1_DC(uint16_t new_duty) // Change the Duty Cycle of the PWM signal
{
 TIM3->CCR1 = new_duty;  //change it to new value//change the speed of motor, update after few
100 seconds 10->1% 990 ->99%
}

/*** Servo Motor at the Exit Gate ***/
void change_CH2_DC(uint16_t new_duty) // Change the Duty Cycle of the PWM signal
{
 TIM3->CCR2 = new_duty;  //change it to new value//change the speed of motor, update after few
100 seconds 10->1% 990 ->99%
}
```