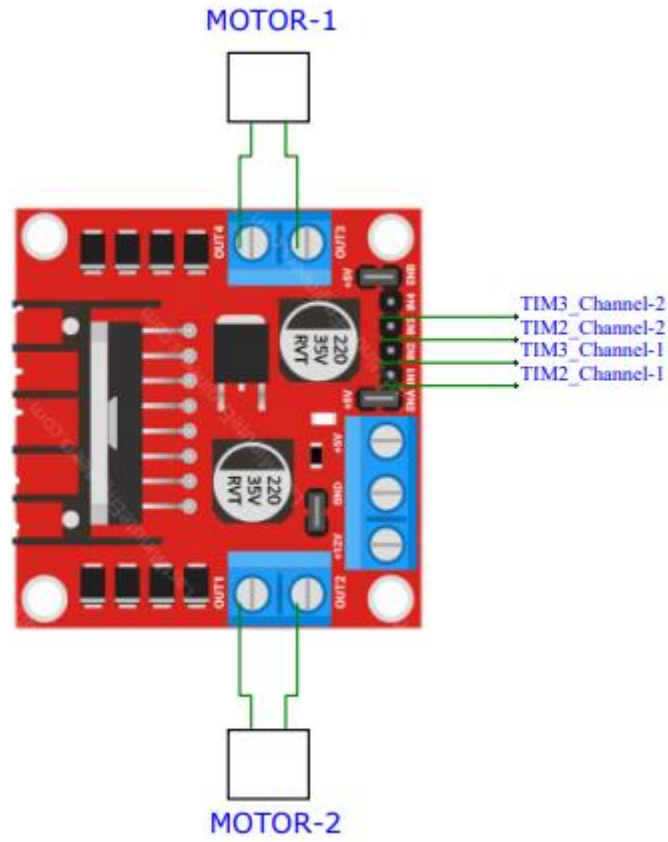
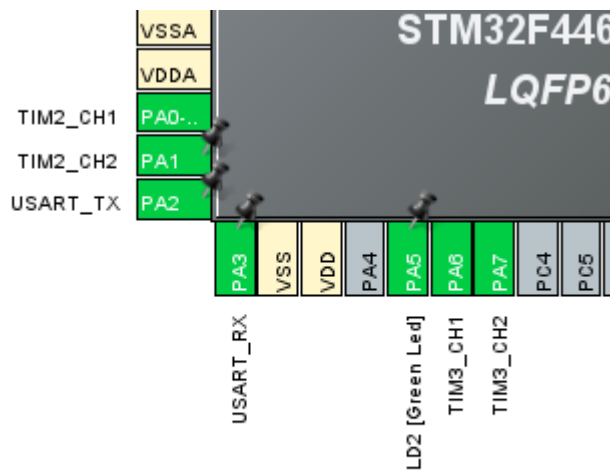


STM32 ile sürücüyü aynı GND hattına bağlamayı unutma



RC aracın kontrol sistemi için biri solda diğeri sağda olmak üzere 2 motor kullanacağım. Bu motorların yönlerini joystickler kullanarak belirleyeceğim.



Şekilde görüldüğü gibi 4 adet timer PWM çıkışını kullanacağım bu bunlardan

TIM2 ayarları : prescaler 41 / counter period 1896

1896 değeri adc max okuma değeri

4096 – 2200 değerinden gelmektedir

TIM3 ayarları : prescaler değeri 41 / counter period değeri 1900

```

typedef struct {
    float coeff[2];
    float out[2];
} RCFilter;

void RCFilter_Init(RCFilter *filt, float cutoffFreqHz, float sampleTimeS);
float RCFilter_Update(RCFilter *filt, float inp);

#endif

```

Rc filtre için RCFilter filter structure oluşturulduktan sonra init edilir

İnit değerleri olarak 0.04f kullandım.

```

25 void motorUpDown(uint32_t *adcValue)
26 {
27     /* if the joystick adc value greater than 2200 it is goes forward direction*/
28     if(adcValue[0] > 2200)
29     {
30         /* if we desire the rotation of car in farward direction we disable this */
31         if(carTurn != FORWARDTURN)
32         {
33             TIM3->CCR1 = 0;
34             TIM3->CCR2 = 0;
35
36             TIM2->CCR1 = RCFilter_Update(&filt, (adcValue[0] - 2200));
37             TIM2->CCR2 = RCFilter_Update(&filt, (adcValue[0] - 2200));
38
39             carDirection = FORWARD;
40
41             filtLeft = filt;
42             filtRight = filt;
43         }
44     }
45     /* if joystick is in idle state reset the PWM values*/
46     else if (adcValue[0] > 1900 && adcValue[0] < 2200)
47     {
48         if(carTurn != STOPTURN)
49         {
50             TIM3->CCR1 = 0;
51             TIM3->CCR2 = 0;
52
53             TIM2->CCR1 = 0;
54             TIM2->CCR2 = 0;

```