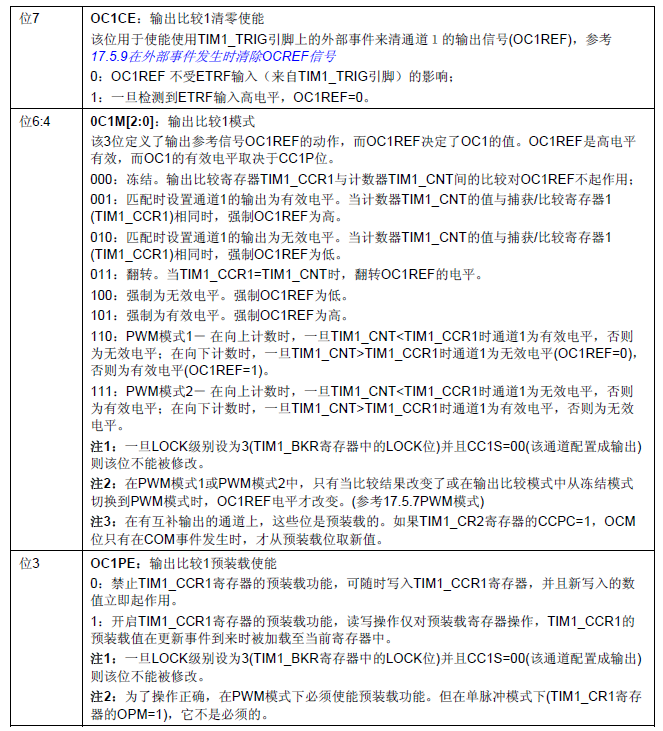
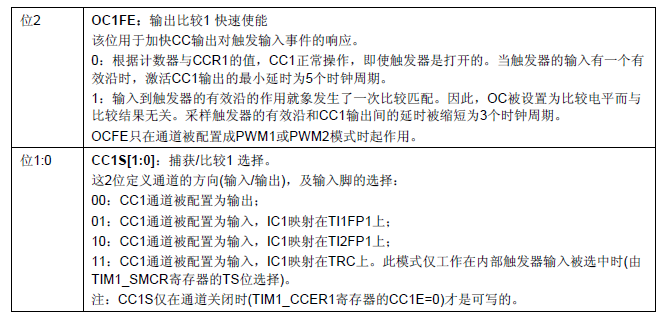


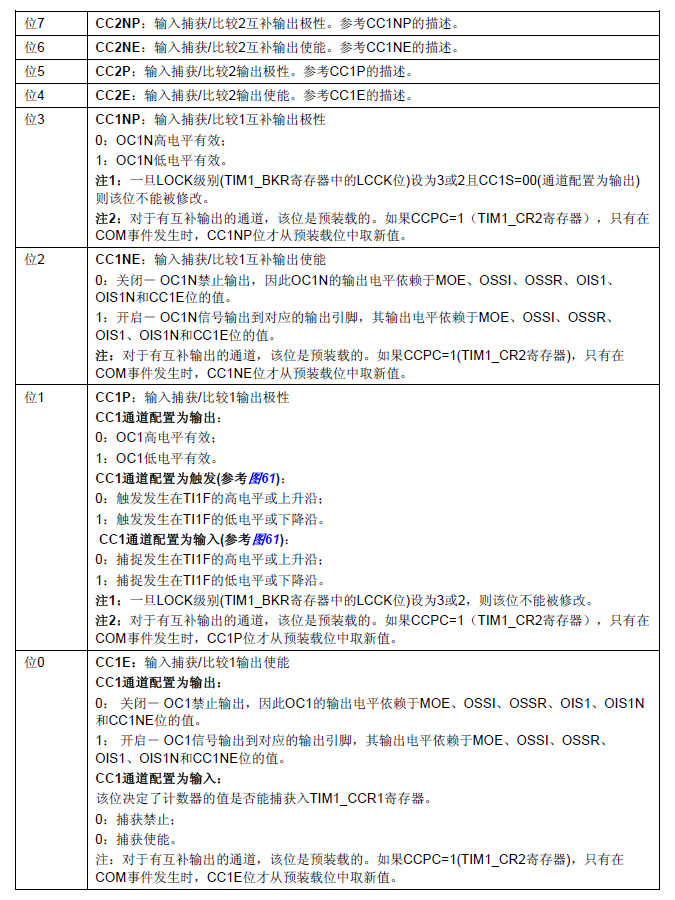
不考虑死区和互补输出，则如图所示，只需要配置：

1. CCMR\_OC1M的输出模式
2. CCER\_CC1NE:CCER\_CC1E = 01;
3. CCER\_CC1P = 高电平，低电平；

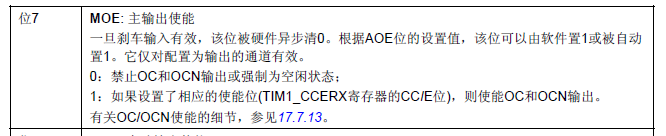
**CCMR1:**

**CCER1:**



**BKR:**



#include "type\_def.h"

void main**(**void**)**

**{**

#if 0

TIM2\_PSCR **=** 0X00**;**

TIM2\_ARRH **=** 0X00**;**

TIM2\_ARRL **=** 136**;**

TIM2\_CR1 **&=** 0XFE**;**

CLK\_CKDIVR**&=** **(**uint8\_t**)(~**0x18**);**/\*使能内部时钟\*/

CLK\_CKDIVR**|=** **(**uint8\_t**)**0x00**;**/\*设置时钟为内部16M高速时钟\*/

TIM2\_CCER1 **&=** 0xFC**;**

TIM2\_CCER1 **|=** 0x01**;**

TIM2\_CCMR1 **&=** 0x8F**;**

TIM2\_CCMR1 **|=** 0X6C**;**

TIM2\_CCR1H **=** 0X00**;**

TIM2\_CCR1L **=** 54**;**

TIM2\_CR1 **|=** 0X01**;**

**while** **(**1**);** //

#else

TIM1\_PSCRH **=** 0X00**;**

TIM1\_PSCRL **=** 0X00**;**

TIM1\_ARRH **=** 0X00**;**

TIM1\_ARRL **=** 136**;**

TIM1\_CR1 **&=** 0XFE**;**

TIM1\_BKR **=** 0X80**;** //这个是TIM1和TIM2的最大的区别

CLK\_CKDIVR**&=** **(**uint8\_t**)(~**0x18**);**/\*使能内部时钟\*/

CLK\_CKDIVR**|=** **(**uint8\_t**)**0x00**;**/\*设置时钟为内部16M高速时钟\*/

TIM1\_CCER1 **&=** 0xFC**;**

TIM1\_CCER1 **|=** 0x01**;**

TIM1\_CCMR1 **&=** 0x8F**;**

TIM1\_CCMR1 **|=** 0X6C**;**

TIM1\_CCR1H **=** 0X00**;**

TIM1\_CCR1L **=** 54**;**

TIM1\_CR1 **|=** 0X01**;**

int i**;**

**while** **(**1**)**

**{**

//TIM1\_CCER1 &= 0xFE; //关闭TIM1输出

**for(**i **=** 0**;**i**<**2000**;**i**++);** //证明可以不影响pwm输出

//TIM1\_CCER1 |= 0x01; //开启TIM1输出

**for(**i **=** 0**;**i**<**2000**;**i**++);**

**}**

#endif

**}**