

## 8.2. Salida por comparación.

Se puede dedicar un pin para sacar una señal cuadrada controlada por la temporización de uno de los contadores. Las siguientes opciones se pueden escoger para tener control sobre la señal de salida:

- Congelado. La comparación entre el registro de comparación y el contador no tiene efecto sobre el pin de salida. Este modo se usa para tener una base de tiempo.
- Salida por comparación activada. La señal de salida se pone en alto cuando el contador es igual al registro de comparación.
- Salida por comparación desactivada. La señal de salida en el pin se pone en nivel bajo cuando el contador es igual al registro de comparación.
- Salida por comparación alternada. La salida de la señal en el pin cambia de estado cada vez que el contador sea igual al registro de comparación.

El procedimiento para obtener una señal de salida con este modo es el siguiente.

1. Seleccionar la fuente de reloj.
2. Escribir los valores en los registros de autocarga (ARR) y de comparación (CCR<sub>x</sub>).
3. Configurar el modo de salida. a) Seleccionar el modo de comparación congelado/ activo/ inactivo o alternado. b) Seleccionar la polaridad. c) Deshabilitar la característica de pre carga, d) Habilitar la salida de captura por comparación.
4. Habilitar el contador.
5. Si hay interrupción o DMA, habilitarlos.

Es decir, la configuración base del apartado 8.1 debe hacerse para el contador y luego hacer la configuración de salida por comparación. El siguiente es un ejemplo de generación de una señal de salida con un led pegado al pin 6 del puerto A, donde está la salida del canal 1 del contador Tim3, generando un cambio a un segundo y otro cambio a dos segundos.

```
Tim3Init();// configuración base del contador

// configuración de salida por comparación del contador 3, canal 1
TIM_OCInitTypeDef TIM_OCInitStruct;
TIM_OCStructInit(&TIM_OCInitStruct);
TIM_OCInitStruct.TIM_OCMode = TIM_OCMode_Toggle;
TIM_OCInitStruct.TIM_OutputState = TIM_OutputState_Enable;
TIM_OCInitStruct.TIM_Pulse = 1000;
TIM_OCInitStruct.TIM_OCPolarity = TIM_OCPolarity_Low;
TIM_OC1Init(TIM3, &TIM_OCInitStruct);
TIM_OC1PreloadConfig(TIM3, TIM_OCPreload_Disable);// la comparación es la misma siempre

// configuración del pin de salida del tim3, canal 1, GPIOA6
RCC_APB2PeriphClockCmd(RCC_APB2Periph_GPIOA, ENABLE);
GPIO_InitTypeDef GPIOB_Struct;
GPIOB_Struct.GPIO_Speed = GPIO_Speed_50MHz;
GPIOB_Struct.GPIO_Mode = GPIO_Mode_AF_PP;
GPIOB_Struct.GPIO_Pin = GPIO_Pin_6;
GPIO_Init(GPIOA, &GPIOB_Struct);
```

En la primera línea se llama la función de configuración del contador 3 (`Tim3Init()`) con dos segundos de periodo y un prescaler de 56.000, para dar un milisegundo por pulso en la entrada del contador. Luego sigue un bloque de configuración de la salida por comparación, escogiendo el modo alterno (`TIM_OCMode_Toggle`), con un registro de comparación en 1.000, o sea, un segundo. Luego sigue el bloque de configuración del pin 6 del puerto A en modo de funciones alternas, pin de salida en push- pull y a alta velocidad.