**STM32F4 Timers**

El micro tiene 17 Timers:

* 12 de 16 bits
* 2 de 32 bits
* 2 Watchdogs
* 1 System Timer

Todos trabajan con un prescaler de 16 bits.

|  |  |
| --- | --- |
| System | Se usa para los RTOS |
| WatchDog | Watchdog |
| Advanced-Control | Timers: 1 y 8 |
| General Propose | Timers: 2, 3, 4, 5, 9, 10, 11, 12, 13 y 14 |
| Basic | Timers: 6 y 7 |

Detalle de cada timer

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Timer** | **Tipo** | **16/32 Bits** | **PWM** | **DMA** | **Encoder** | **Info** |
| Timer1 | A-C | 16 | Si | Si |  | up/down counter |
| Timer2 | GP | **32** | Si | Si | Si | up/down counter |
| Timer3 | GP | 16 | Si | Si | Si | up/down counter |
| Timer4 | GP | 16 | Si | Si | Si | up/down counter |
| Timer5 | GP | **32** | Si | Si | Si | up/down counter |
| Timer6 | Basic | 16 |  | Si |  | Se usan para el DAC, up counter |
| Timer7 | Basic | 16 |  | Si |  | Se usan para el DAC, up counter |
| Timer8 | A-C | 16 | Si | Si |  | up/down counter |
| Timer9 | GP | 16 |  |  |  | Solo upcounter |
| Timer10 | GP | 16 |  |  |  | Solo upcounter |
| Timer11 | GP | 16 |  |  |  | Solo upcounter |
| Timer12 | GP | 16 |  |  |  | Solo upcounter |
| Timer13 | GP | 16 |  |  |  | Solo upcounter |
| Timer14 | GP | 16 |  |  |  | Solo upcounter |
| Independent Watchdog |  | 12, prescaler: 8 bits |  |  |  |  |
| Window Watchdog |  | 7 bits |  |  |  |  |
| Sys Tick |  | 24 bits |  |  |  | down counter |

**Advanced - Control Timers**

1. Especificaciones

Son dos timers, T1 y T8, de 16 bits, pueden ser usados para generar PWM, el hardware es completamente independiente de los GP Timers y puede sincronizarse con ellos. Soportan DMA para una cantidad de eventos y pueden ser usados para leer los Quadrature Encoders.

Tienen 4 canales:

* Input Capture
* Output Compare
* PWM Generation
* One Pulse mode output

2. Funcionamiento

El timer utiliza una serie de registros:

|  |  |  |
| --- | --- | --- |
| **Registro** | **Nombre** | **Descripcion** |
| TIMx\_CNT | Counter Register | Cuenta los ticks del timer hasta el valor del ARR, despues genera el evento UEV y comienza desde cero nuevamente (en up counting mode) |
| TIMx\_PSC | Prescaler Register | Divisor de la frecuencia de entrada del timer. |
| TIMx\_ARR | Auto Reload Register |  |
| TIMx\_RCR | Repetition Counter Register | Retarda la generacion del evento UEV la cantidad de veces que indique el valor de este registro. Es decir, en lugar de generarse un evento por cada overflow, se genera uno por cada RCR overflows + 1. |
| TIMx\_SR |  | Configuracion |
| TIMx\_CR2 |  | Configuracion |

Eventos:

|  |  |  |
| --- | --- | --- |
| **Evento** | **Nombre** | **Descripcion** |
| UEV | Update Event | Evento de actualizacion, se dispara cuando el contador llega overflow (under o over) se setea el bit UIF en el registro SR |

Registros de control:

* CNT\_EN: Habilita el timer.
* UIF: Indica si hubo un UEV

La entrada del clock puede ser tomada de distintas fuentes:

* Internal Clock (CK\_INT)
* External clock mode 1: external input pin
* Otro timer que actue como prescaler del primero. (El timer 1 puede actuar como prescaler del timer2)

3. Canales de salida

Al parecer, los canales de un timer son contadores que trabajan cada uno con una lógica independiente del otro. De esta forma se puede evaluar una señal con un timer de varias formas distintas en paralelo.

Los timers del micro tienen muchos modos de operación, algunos bastante estrambóticos.

3.1 PWM Mode

Genera una señal con una frecuencia determinada por el registro ARR y un duty cycle determinado por el registro CCR

# Anotaciones:

La función SystemInit () se corre al principio de todo. El código esta en el archivo assembler, después de correr eso se llama a main().

Para generar un PWM hay que seleccionar un Pin de salida del micro para poner la señal. El tema con esos pines es que son compartidos por muchos periféricos, entonces, hay que primero asignarle al periférico (timer) un pin determinado para que saque la señal del PWM por ahí. Esto se hace configurándolo como alternate function. Ver pag 189.