Chapter 47 Timer

47.1 Overview

Timer is a programmable timer peripheral. This component is an APB slave device.

Timer0~4 and Timer6 count down from a programmed value and generate an interrupt when the count reaches zero.

Timer5 and Timer7 count up from zero to a programmed value and generate an interrupt when the count reaches the programmed value.

Timer supports the following features:

- Two APB timers in the soc system. One is in the alive subsystem, include two programmable 64 bits timer channel, acts as TIMER6 and TIMER7; The other is in the cpu subsystem, include six programmable 64 bits timer channel, acts as TIMER0, TIMER1, TIMER2, TIMER3, TIMER4, and TIMER5 respectively.
- Two operation modes: free-running and user-defined count.
- Maskable for each individual interrupt.
- TIMER5 is used for gpu ; TIMER7 is used for core.

47.2 Block Diagram

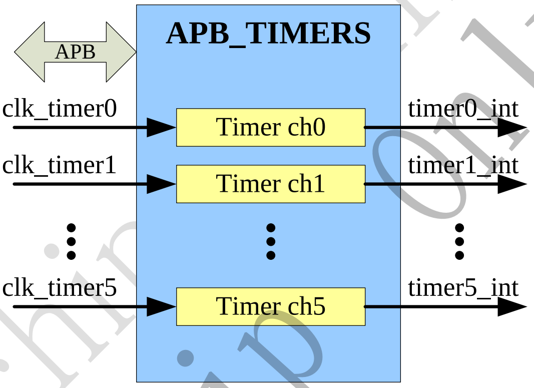

Fig. 47-1 Timers Block Diagram

The above figure shows the architecture of the APB timers (include six programmable timer channel) that in the cpu subsystem. The other APB timers that in the alive subsystem only include one programmable timer channel.

47.3 Function description

47.3.1 Timer clock

TIMER0, TIMER1, TIMER2, Timer3, TIMER4 and TIMER5 are in the CPU subsystem, using timer ch0 ~ ch5 respectively. The timer clock is 24MHz OSC.

TIMER6 and TIMER7 are in the ALIVE subsystem, using timer ch0 ~ ch1. The timer clock is 24MHz OSC.

47.3.2 Programming sequence

1. Initialize the timer by the TIMERn_CONTROLREG (0≤n≤5) register:

- Disable the timer by writing a “0” to the timer enable bit (bit 0). Accordingly, the timer_en output signal is de-asserted.

- Program the timer mode—user-defined or free-running—by writing a "0" or "1" respectively, to the timer mode bit (bit 1).
- Set the interrupt mask as either masked or not masked by writing a "0" or "1" respectively, to the timer interrupt mask bit (bit 2).

2. Load the timer count value into the TIMERn_LOAD_COUNT1 (0⩽n⩽5) and TIMERn_ LOAD_COUNT0 (0⩽n⩽5) register.

3. Enable the timer by writing a "1" to bit 0 of TIMERn_CONTROLREG (0⩽n⩽5).

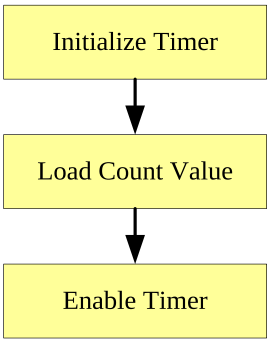

Fig. 47-2 Timer Usage Flow

47.3.3 Loading a timer count value

The initial value for each timer — that is, the value from which it counts down — is loaded into the timer using the load count register (TIMERn_LOAD_ COUNT1 (0⩽n⩽5) and TIMERn_ LOAD_COUNT0 (0⩽n⩽5)). Two events can cause a timer to load the initial value from its load count register:

- Timer is enabled after reset or disabled.
- Timer counts down to 0, when timer is configured into free-running mode.

47.3.4 Timer mode selection

- User-defined count mode – Timer loads TIMERn_LOAD_COUNT1 (0⩽n⩽5) and TIMERn_ LOAD_ COUNT0 (0⩽n⩽5) register as initial value. Timer will not automatically load the count register, when timer counts down to 0. User need to disable timer firstly and follow the programming sequence to make timer work again.
- Free-running mode – Timer loads the TIMERn_LOAD_COUNT1 (0⩽n⩽5) and TIMERn_ LOAD_ COUNT0 (0⩽n⩽5) register as initial value. Timer will automatically load the count register, when timer counts down to 0.

47.4 Register Description

This section describes the control/status registers of the design. Software should read and write these registers using 32-bits accesses.

47.4.1 Registers Summary

Name	Offset	Size	Reset Value	Description
TIMER0_LOAD_COUNT0	0x0000	W	0x00000000	Timer0 Load Count Register
TIMER0_LOAD_COUNT1	0x0004	W	0x00000000	Timer0 Load Count Register
TIMER0_CURRENT_VALUE0	0x0008	W	0x00000000	Timer0 Current Value Register
TIMER0_CURRENT_VALUE1	0x000C	W	0x00000000	Timer0 Current Value Register
TIMER0_CONTROLREG	0x0010	W	0x00000000	Timer0 Control Register

TIMER0_INTSTATUS	0x0018	W	0x00000000	Timer0 Interrupt Status Register
TIMER1_LOAD_COUNT0	0x0020	W	0x00000000	Timer1 Load Count Register
TIMER1_LOAD_COUNT1	0x0024	W	0x00000000	Timer1 Load Count Register
TIMER1_CURRENT_VALUE0	0x0028	W	0x00000000	Timer1 Current Value Register
TIMER1_CURRENT_VALUE1	0x002c	W	0x00000000	Timer1 Current Value Register
TIMER1_CONTROLREG	0x0030	W	0x00000000	Timer1 Control Register
TIMER1_INTSTATUS	0x0038	W	0x00000000	Timer1 Interrupt Status Register
TIMER2_LOAD_COUNT0	0x0040	W	0x00000000	Timer2 Load Count Register
TIMER2_LOAD_COUNT1	0x0044	W	0x00000000	Timer2 Load Count Register
TIMER2_CURRENT_VALUE0	0x0048	W	0x00000000	Timer2 Current Value Register
TIMER2_CURRENT_VALUE1	0x004c	W	0x00000000	Timer2 Current Value Register
TIMER2_CONTROLREG	0x0050	W	0x00000000	Timer2 Control Register
TIMER2_INTSTATUS	0x0058	W	0x00000000	Timer2 Interrupt Status Register
TIMER3_LOAD_COUNT0	0x0060	W	0x00000000	Timer3 Load Count Register
TIMER3_LOAD_COUNT1	0x0064	W	0x00000000	Timer3 Load Count Register
TIMER3_CURRENT_VALUE0	0x0068	W	0x00000000	Timer3 Current Value Register
TIMER3_CURRENT_VALUE1	0x006c	W	0x00000000	Timer3 Current Value Register
TIMER3_CONTROLREG	0x0070	W	0x00000000	Timer3 Control Register
TIMER3_INTSTATUS	0x0078	W	0x00000000	Timer3 Interrupt Status Register
TIMER4_LOAD_COUNT0	0x0080	W	0x00000000	Timer4 Load Count Register
TIMER4_LOAD_COUNT1	0x0084	W	0x00000000	Timer4 Load Count Register
TIMER4_CURRENT_VALUE0	0x0088	W	0x00000000	Timer4 Current Value Register
TIMER4_CURRENT_VALUE1	0x008c	W	0x00000000	Timer4 Current Value Register
TIMER4_CONTROLREG	0x0090	W	0x00000000	Timer4 Control Register
TIMER4_INTSTATUS	0x0098	W	0x00000000	Timer4 Interrupt Status Register
TIMER5_LOAD_COUNT0	0x00a0	W	0x00000000	Timer5 Load Count Register
TIMER5_LOAD_COUNT1	0x00a4	W	0x00000000	Timer5 Load Count Register
TIMER5_CURRENT_VALUE0	0x00a8	W	0x00000000	Timer5 Current Value Register
TIMER5_CURRENT_VALUE1	0x00ac	W	0x00000000	Timer5 Current Value Register
TIMER5_CONTROLREG	0x00b0	W	0x00000000	Timer5 Control Register
TIMER5_INTSTATUS	0x00b8	W	0x00000000	Timer5 Interrupt Status Register

*Notes: Size: **B** – Byte (8 bits) access, **HW** – Half WORD (16 bits) access, **W** –WORD (32 bits) access*

47.4.2 Detail Register Description

TIMERn_LOAD_COUNT0

Address: Operational Base + offset(0x00+n*0x20)

Timer n Load Count Register (0≤n≤5)

Bit	Attr	Reset Value	Description
31:0	RW	0x0	Low 32 bits Value to be loaded into Timer n. This is the value from which counting commences.

TIMERn_LOAD_COUNT1

Address: Operational Base + offset(0x04+n*0x20)

Timer n Load Count Register (0≤n≤5)

Bit	Attr	Reset Value	Description
31:0	RW	0x0	High 32 bits Value to be loaded into Timer n. This is the value from which counting commences.

TIMERn_CURRENT_VALUE0
Address: Operational Base + offset(0x08+n*0x20)
Timer n Current Value Register (0≤n≤5)

Bit	Attr	Reset Value	Description
31:0	R	0x0	Low 32 bits of Current Value of Timer n.

TIMERn_CURRENT_VALUE1
Address: Operational Base + offset(0x0c+n*0x20)
Timer n Current Value Register (0≤n≤5)

Bit	Attr	Reset Value	Description
31:0	R	0x0	High 32 bits of Current Value of Timer n.

TIMERn_CONTROLREG
Address: Operational Base + offset(0x10+n*0x20)
Timer n Control Register (0≤n≤5)

Bit	Attr	Reset Value	Description
31:3	-	-	Reserved
2	RW	0x0	Timer interrupt mask. 1'b0: mask 1'b1: not mask
1	RW	0x0	Timer mode. 1'b0: free-running mode 1'b1: user-defined count mode
0	RW	0x0	Timer enable. 1'b0: disable 1'b1: enable

TIMERn_INTSTATUS
Address: Operational Base + offset(0x18+n*0x20)
Timer n Interrupt Status Register (0≤n≤5)

Bit	Attr	Reset Value	Description
31:1	-	-	Reserved
0	RW	0x0	This register contains the interrupt status for timer n Write 1 to this register will clear the interrupt

*Notes: Attr: **RW** – Read/writable, **R** – read only, **W** – write only*

47.5 Application Notes

In the chip, the timer_clk is from 24MHz OSC, asynchronous to the pclk. When user disable the timer enable bit (bit 0 of TIMERn_CONTROLREG (0≤n≤5)), the timer_en output signal is de-asserted, and timer_clk will stop. When user enable the timer, the timer_en signal is asserted and timer_clk will start running.

The application is only allowed to re-config registers when timer_en is low.

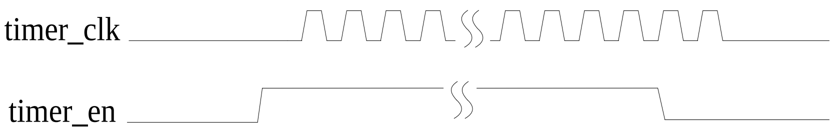

Fig. 47-3 Timing between timer_en and timer_clk

Please refer to funciton description section for the timer usage flow.