

LECTURE 27 - TIMER

Q. What is timer?

- A.
- ① Timer is fully depend upon the oscillator that attached externally to the microcontroller because it uses the frequency of oscillator to operate.
 - ② When we trigger timer it starts from initial value and run up to decided value stored by user in special function registers.
 - ③ When it reaches its maximum value, it overflows and a certain bit is decided to show overflow in SFR (special function register) also called flag bit.
 - ④ Some timers also used prescalar technique to enhance its limits. For example,

8-bit timer : $256 (2^8)$

16-bit timer : $65536 (2^{16})$

32-bit timer : 2^{32}

+ TIMER IN LPC2148:

- ① The LPC2148 has two functionally identical general purpose timers, Timer0 and Timer1. Both timer/counter with a programmable 32-bit prescaler; counter/timer operation.
- ② Up to four 32-bit capture channels per timer, that can take a snapshot of the timer value when an input signal transitions. A capture event may also optionally generate an interrupt.
- ③ Four 32-bit match registers that allow-
 - (i) continuous operation with optional interrupt generation on match.
 - (ii) stop timer on match with optional interrupt generation.
 - (iii) Reset timer on match with optional interrupt generation.

- ① Up to four external outputs corresponding to match registers, with the following capabilities -
- Set low on match.
 - Set high on match.
 - Toggle on match.
 - Do nothing on match.

* CAPTURE REGISTER:

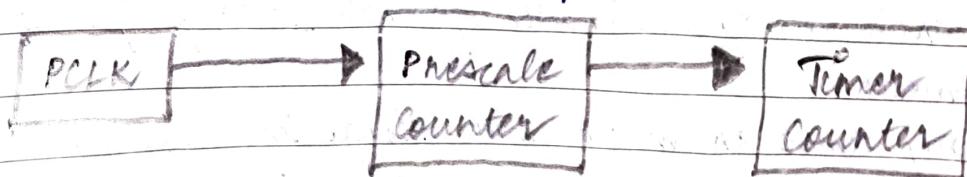
As the name suggests it is used to capture input signal. When a transition event occurs on a capture pin, it can be used to copy the value of TC into any of the 4 Capture Register or to generate an interrupt. Hence, these can be also used to demodulate PWM signals.

* MATCH REGISTER:

A match register is a register which contains a specific value set by the user. When the timer starts, every time after TC is incremented, the value in TC is compared with match register. If it matches then it can reset the timer or can generate an interrupt as defined by the user. We are only concerned with match registers in this lecture.

Q. How timers in LPC2148 ARM1 microcontroller works?

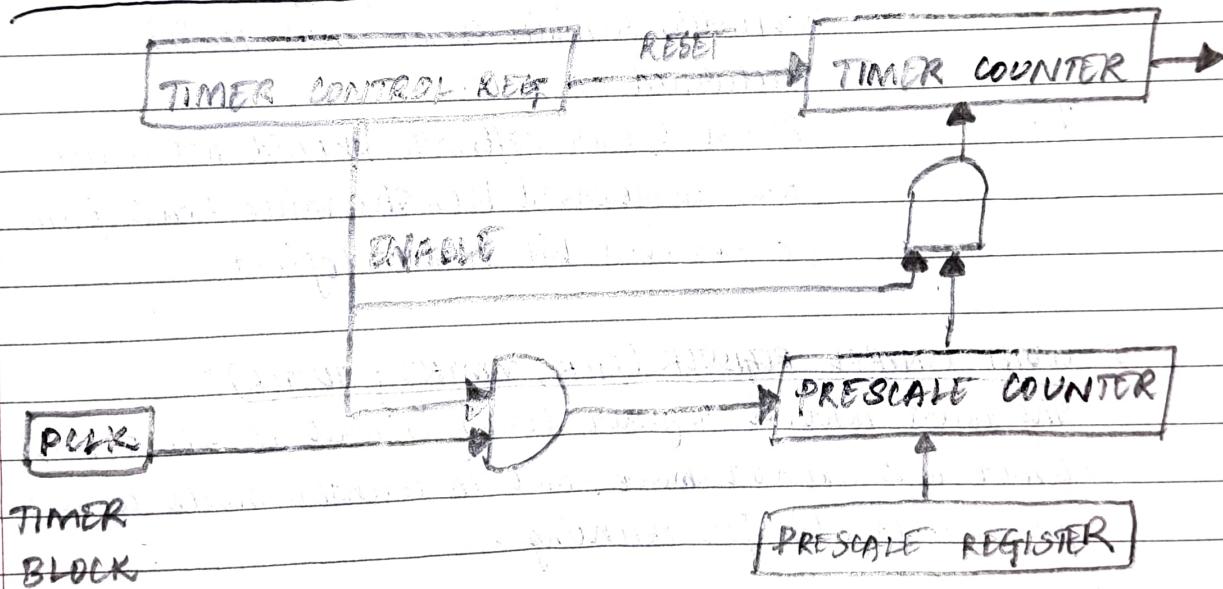
- A. The heart of timers of the LPC2148 microcontroller is a 32-bit free running counter, which is designed to count cycles of the peripheral clock (PCLK) or an external clock, this counter is programmable with 32-bit prescaler.



The tick rate of the timer counter (TC) is controlled by the 32-bit number written in the prescale register (PR) in the following way-

- (i) There is a prescale counter (PC) which increments on each tick of the PCLK.
- (ii) When it reaches the value in the prescaler register, the timer count is incremented and the prescaler counter (PC) is reset, on the next PCLK.
- (iii) This cause the timer counter to increment on every PCLK when $PR \neq 0$, every 2 PCLKs when $PR = 1$, etc.

* LPC2148 TIMER BLOCK:



The prescale counter is incremented on every PCLK. When prescale counter reaches the value stored in the prescale register, the timer counter is incremented and the prescale counter is reset on the next PCLK.

The match register values are continuously compared to the timer counter value, when the two values are equal, actions (match pin / interrupt) can be triggered automatically.

* TIMER CONTROL REGISTER (TCR, TIMERO : TDTCR) :

Timer control register used to control the timer control functions. We'll enable, disable and reset timer counter (TC) through this register.

Bit	Symbol	Description	Reset value
0	Counter Enable	When one, the timer counter and prescale counter are enabled for counting. When zero, the counters are disabled.	0
1	Counter Reset	When one, the timer counter and the prescale counter are synchronously reset on the next positive edge of PCLK. The counters remain reset until TCR[1] is returned to zero.	0
7:2	-	Reserved, user software should not write ones to reserved bits. The value read from a reserved bit is not defined.	NA

* COUNT CONTROL REGISTER (CTCR, TIMERO : TDCTCR) :

The count control register (CTCR) is used to select between timer and counter mode, and in counter mode to select the pin and edge(s) for counting.

Bit	Symbol	Value	Description
1 ⁰	Counter/Timer Mode		This field selects which rising PCLK edges can increment timer's prescale counter (PC), or clear PC and increment timer counter by
00		00	Timer Mode: every rising PCLK edge
01		01	Counter Mode: TC is incremented on rising edges on the CAP input selected by bits 3:2
10		10	Counter mode: TC is incremented on falling edges on the CAP input selected by bits 3:2

II Counter mode: TC is incremented on both edges on the CAP input selected by bits 3:2.

* TIMER COUNTER (TC, TIMERO : TOCR):

This is the main counting register. Timer counter increments when PC reaches its maximum value as specified by PR. If timer is not reset explicitly (directly) or by using an interrupt then it will act as a free running counter which resets back to zero when it reaches its maximum value which is 0xFFFF FFFF.

* PRESCALE REGISTER (PR, TIMERO : TDPR):

The 32-bit prescale register specifies the maximum value for the prescale counter.

* PRESCALE COUNTER REGISTER (PC, TIMERO : TDPC):

- This register increments on every PCLK (peripheral clock). This register controls the resolution of the timer. When PC reaches the value in PR, PC is reset back to 0 and timer counter is incremented by 1.
- Hence if $PR=0$ then timer counter increments on every 1 PCLK. If $PR=9$, then timer counter increments on every 10th cycle of PCLK. Hence, by selecting an appropriate prescale value we can control the resolution of the timer.

* MATCH REGISTERS (MRO - MR3):

The match register values are continuously compared to the timer counter value. When the two values are equal, actions are triggered automatically. The action possibilities are to generate an interrupt, reset the timer counter, or stop the timer. Actions are controlled by the settings in the MCR register.

* MATCH CONTROL REGISTER : (MCR, TIMER0 : TDMCR)

This register is used to control which all operations can be done when the value in MR matches the value in TC. Bits 0, 1, 2 are for MRO, Bits 3, 4, 5 for MRI and so on. For MRO (Match Register 0):-

- ① Bit 0 - Interrupt on MRO i.e. trigger an interrupt when MRO matches TC. Interrupts are enabled when set to 1 and disabled when set to 0.
- ② Bit 1 - Reset on MRO. When set to 1, TC will be reset when it matched MRO. Disabled when set to 0.
- ③ Bit 2 - Stop on MRO. When set to 1, TC and PC will stop when MRO matches TC.
- ④ Similarly bits 3-5, 6-8, 9-11 are for MRI, MR2, MR3 respectively.

* INTERRUPT REGISTER (IR, TIMER0 : TDIR):

The interrupt register consists of four bits for the match interrupts and four bits for the capture interrupts.

Bit	Symbol	Description
0	MRO Interrupt	Interrupt flag for match channel 0.
1	MRI Interrupt	Interrupt flag for match channel 1.
2	MR2 Interrupt	Interrupt flag for match channel 2.
3	MR3 Interrupt	Interrupt flag for match channel 3.
4	CRO Interrupt	Interrupt flag for capture channel 0 event.
5	CRI Interrupt	Interrupt flag for capture channel 1 event.
6	CR2 Interrupt	Interrupt flag for capture channel 2 event.
7	CR3 Interrupt	Interrupt flag for capture channel 3 event.

* PRESCALE (TxPR) RELATED CALCULATIONS:

The delay or time required for 1 clock cycle at 'x' MHz is given by - $\frac{1}{x \times 10^6}$ seconds.

Hence, in our case when PR=0 i.e. TC increments at every PCLK the delay required per TC to increment by 1 is -

$$\frac{0+1}{60 \times 10^6} \text{ seconds}$$

Similarly when we set PR = 59999, the delay in this case will be -

$$\frac{59999+1}{60 \times 10^6} = \frac{60000}{60 \times 10^6} \times 1000 = \frac{1}{10^3} \text{ seconds}$$

which boils down to 0.001 seconds which is nothing but 1 milli-second i.e. ms. Hence, the delay required for TC to increment by 1 will be 1 ms.

* BASIC STEP FOR TIMER :

- ① Set appropriate value in TxCTCR.
- ② Define the prescale value in TxPR.
- ③ Set value(s) in match register(s), if required.
- ④ Set appropriate value in TrMCR if using match register/interrupt.
- ⑤ Reset timer, which resets PR and TC.
- ⑥ Set TxTCR to 0x01 to enable the timer when required.
- ⑦ Reset TxTCR to 0x00 to disable the timer when required.

Note - x denotes which timer we want to use.