



# TM57 Series

TM57PE40 DEMO CODE FOR  
TM57PE40  
BASIC FUNCTIONS SAMPLE

## Application Note

tenx reserves the right to change or discontinue the manual and online documentation to this product herein to improve reliability, function or design without further notice. tenx does not assume any liability arising out of the application or use of any product or circuit described herein; neither does it convey any license under its patent rights nor the rights of others. tenx products are not designed, intended, or authorized for use in life support appliances, devices, or systems. If Buyer purchases or uses tenx products for any such unintended or unauthorized application, Buyer shall indemnify and hold tenx and its officers, employees, subsidiaries, affiliates and distributors harmless against all claims, cost, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use even if such claim alleges that tenx was negligent regarding the design or manufacture of the part.

**AMENDMENT HISTORY**

<b>Version</b>	<b>Date</b>	<b>Description</b>
V1.0	June, 2011	New release

**CONTENTS**

**AMENDMENT HISTORY ..... 1**

**Product Name..... 3**

**Title..... 3**

**01. CMP Application Sample ..... 3**

**02. PWM Application Sample ..... 5**

**03. Timer0 Application Sample ..... 6**

**04. Timer1 Application Sample ..... 9**

**05. TIMER2 Application Sample..... 11**

**06. WDT/WKT/XINT Application Sample..... 13**

**07. TK Application Sample ..... 14**

**08. TCOU Application Sample..... 17**

**Product Name**

TM57 Series IC

**Title**

TM57PE40\_CMP Application Sample  
 TM57PE40\_PWM Application Sample  
 TM57PE40\_Timer0 Application Sample  
 TM57PE40\_Timer1 Application Sample  
 TM57PE40\_Timer2 Application Sample  
 TM57PE40\_WKT/WDT/XINT Application Sample  
 TM57PE40\_TK Application Sample  
 TM57PE40\_TCOUT Application Sample

**01. CMP Application Sample**

1. Sample function: Compare PD0(in0-) and PD1(in+) voltage, with PD2 output result. Details of DEMO routine please refer to CMP.asm.

**2. CMP related registers****F-Plane**

Address	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
06H	pbd7	pbd6	pbd5	pbd4	pbd3	pbd2	pbd1	pbd0
08H	cmpie	Tm2ie	Tm1ie	Tm2ie	wktie	Xint2e	Xint1e	Xint0e
09H	cmpif	Tm2if	Tm1if	Tm2if	wktif	Xint2if	Xint1if	Xint0if
0DH	pwmadtl7	pwmadtl6	-	-	-	-	-	cmpst
0FH	-	Sircsel1	Sircsel0	stpck	Selsub	Sube	Subtyp1	Subtyp0

**R-Plane**

Address	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
06H	Pbe7	Pbe6	Pbe5	Pbe4	Pbe3	Pbe2	pbe1	pbe0
09H	Npbpu7	npbpu6	npbpu5	npbpu4	npbpu3	npbpu2	npbpu1	npbpu0
0Eh	Wdtpsc1	Wdtpsc0	wdtslpstp	Tm2clks	Fircsel1	Fircsel0	Tm2psc1	Tm2psc0
0fH	cmpen	cmpedge	cmpoe	cmpinns	Cmpinps3	Cmpinps2	Cmpinps1	Cmpinps0

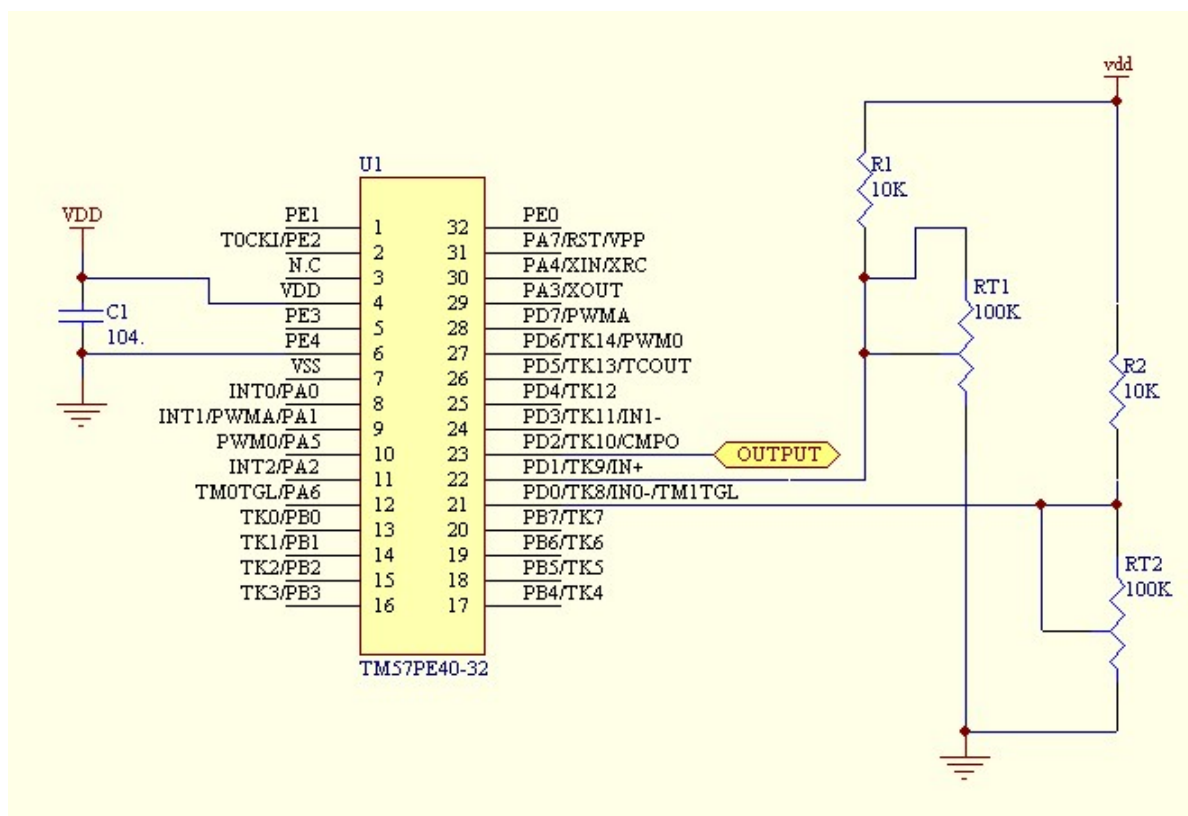
### 3. Sample description

**3-1.** TM57PE40 comparator can output directly to PD2 port, and also produce interrupt. Positive input of comparator contains internal voltage and PD1, while negative input contains 2 ways, i.e. PD0 and PD3.

**3-2.** CMP configuration:

- (1) Set PD2 as output port (set PDE2 to 1), PD1/PD0 as input port (set PDE1~0 to 0, PDD1~0 to 1), close PD1/PD0 pull high (set PDPU1, PDPU0 to 1).
- (2) Set CMPCTRL (r\_plane,10H.7~0): enable comparator (cmpen=1, r\_plane,10H.7), PD2 enable output (cmpoe=1, r\_plane,10H.5), negative input source selection (cmpinns =0, r\_plane,10H.4, in this case is PD0), positive input source selection (cmpinps =1111b, r\_plane,10H.3~0, 本例为 PD1).
- (3) When PD0 voltage is higher than PD1, PD2 will output low. Otherwise, PD2 will output high.

### 4. Circuit diagram



## 02. PWM Application Sample

1. Sample function: After power on, PA1 (PWMA) output periodic cycle is 64 us, the duty is 1/2 waveform, PA5 (PWM0) output periodic cycle is 64 us, duty is 1/2 waveform. Details of the DEMO routine please refer to PWMA.asm and PWM0.asm.

### 2. PWMA/0 related registers

#### F-Plane

Address	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
05H	Pad7	Pad6	Pad5	Pad4	Pad3	Pad2	Pad1	Pad0
07H	Pdd7	Pdd6	Pdd5	Pdd4	Pdd3	Pdd2	Pdd1	Pdd0
0CH	Pwmadth7	Pwmadth6	Pwmadth5	Pwmadth4	Pwmadth3	Pwmadth2	Pwmadth1	Pwmadth0
0DH	Pwmadtl1	Pwmadtl0	-	-	-	-	-	-
0EH	Pwm0duty7	Pwm0duty6	Pwm0duty5	Pwm0duty4	Pwm0duty3	Pwm0duty2	Pwm0duty1	Pwm0duty0
0FH	-	Sircsel1	Sircsel0	stpck	Selsub	Sube	Subtyp1	Subtyp0

#### R-Plane

Address	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
05H	Pae7	Pae6	Pae5	Pae4	Pae3	Pae2	Pae1	Pae0
07H	Pde7	Pde6	Pde5	Pde4	Pde3	Pde2	Pde1	Pde0
0BH	Pwm0psc1	Pwm0psc0	pwm0ae	Pwm0e	Pwm0inv	Tm0out	Wktpsc1	Wktpsc0
0EH	Wdtpsc1	Wdtpsc0	wdtstp	Tm2clks	Fircsel1	Fircsel0	Tm2psc1	Tm2psc0
17H	-	-	-	-	-	-	Pwma_pd7	Pwm0_pd6

### 3. Sample description

- 3-1.** PWM0 is an 8-bit PWM, the clock source is system clock Fcpuclk. User can configure period (PWM0PROD) and duty (PWM0DUTY), and also can select output port PA5 and PD6. PWMA is a 10-bit PWM, the clock source is system clock Fcpuclk. User can configure only the duty (PWMADTH and PWMADTL), and also the output port PA1 and PD7.

- (1) PWM0 duty calculation equation:

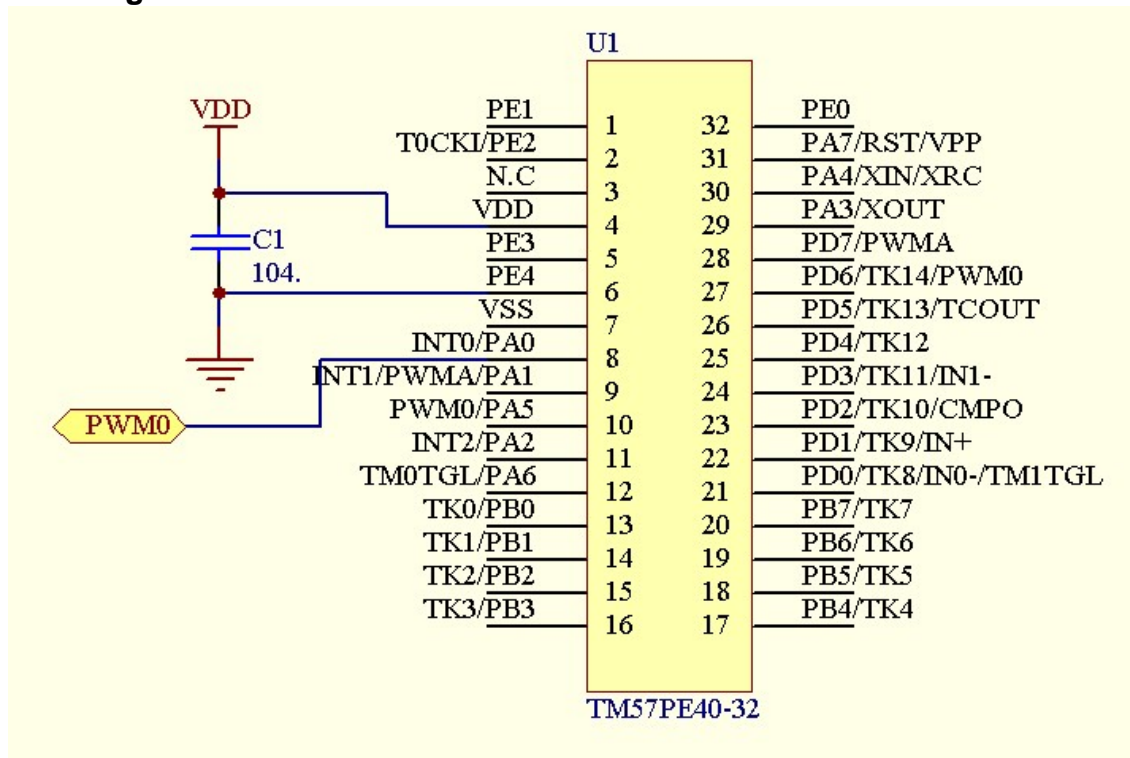
$$\text{PWM0 DUTY OUTPUT} = \text{PWM0DUTY} / (\text{PWM0PROD} + 1)$$

Example: PWM0DUTY is 80h, PWM0PROD is ffh, the output duty will be 50%.

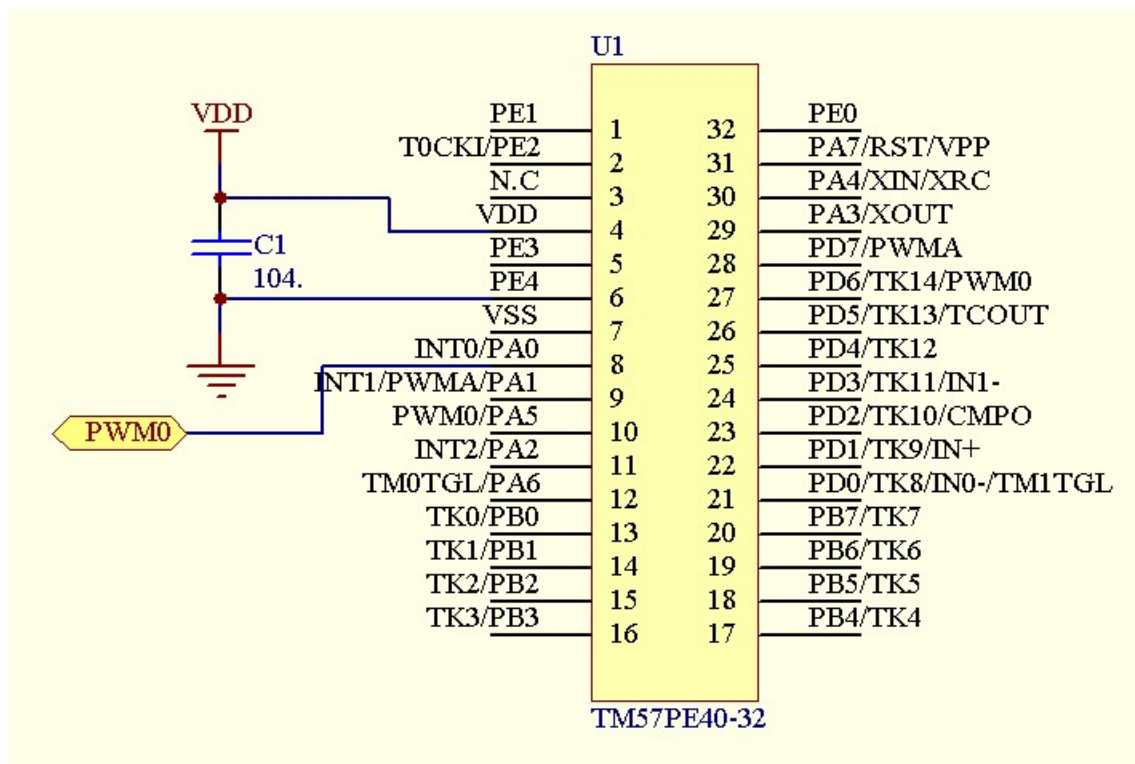
**3-2.** PWM output configuration steps:

- (1) Set PA1 (Pae1=1, r\_plane, 05H.1), PA5 (Pae5=1, r\_plane, 05H.5) port as output.
- (2) Set duty of PWMA/0 (pwmadth: f\_plane, 0CH.7~0; pwmadtl: f\_plane, 0DH.7~6; pwm0duty: f\_plane, 0EH.7~0) and PWM0PROD value of PWM0 (r\_plane, 11H.7~0).
- (3) Select PWMA/0 frequency division (pwm0: r\_plane, 0BH.7~6; pwma has no frequency division, therefore can only use system clock).
- (4) Select PWMA/0 output port, which are PA1 (Pwma\_pd7=0, r\_plane, 17H.1) and PA5 (Pwma\_pd6=0, r\_plane, 17H.0).
- (5) Enable PWMA/0 output, PWMA output enable (pwm0ae=1, r\_plane, 0BH.5), PWM0 output enable (pwm0e=1, r\_plane, 0BH.4).

## 4. Circuit diagram



PWMA circuit diagram



PWM0 circuit diagram

### 03. Timer0 Application Sample

1. Sample function: After power on, the LED output frequency is 500 Hz. Details of DEMO routine please refer to Timer0.asm.

#### 2. Timer0 related registers

F-Plane

Address	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
01H	Timer0 content							
08H	cmpie	Tm2ie	tm1i	tm0ie	wktie	xint2ie	xint1ie	xint0ie
09H	cmpif	Tm2if	Tm1if	Tm0if	wktif	Xint2i	Xint1i	Xint0i

R-Plane

Address	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
02H	capola	T0cap	t0iedge	selt0i	tm0psc3	tm0psc2	tm0psc1	tm0psc0
0Eh	Wdtpsc1	Wdtpsc0	wdtstp	Tm2clks	Fircsel1	Fircsel0	Tm2psc1	Tm2psc0

#### 3. Sample description

**3-1.** Timer0 is an 8-bit increasing timer, when timer counts from 0xFF to 0, overflow occurs, timer interrupts. It does not support reload function, therefore, after interrupt, TIMER0 initial value needs to be reset. TIMER0 clock source option is decided by SELT0I, if SELT0I=0, the clock source is fosc/2, SELT0I clock source is input from T01 pin.

**3-2.** Timer time calculation equation:

$$\text{Time} = 1/\text{fosc} * 2 * \text{Timer0 Pre-Scale} * (256 - \text{TIMER0})$$

External counter calculation equation:

$$\text{Tcount} = 1/\text{Ftoi} * \text{Timer0 Pre-Scale} * (256 - \text{TIMER0}) \quad (\text{Ftoi is T0i input frequency})$$

Use DEMO as sample, timer 1 ms calculation is as follows:

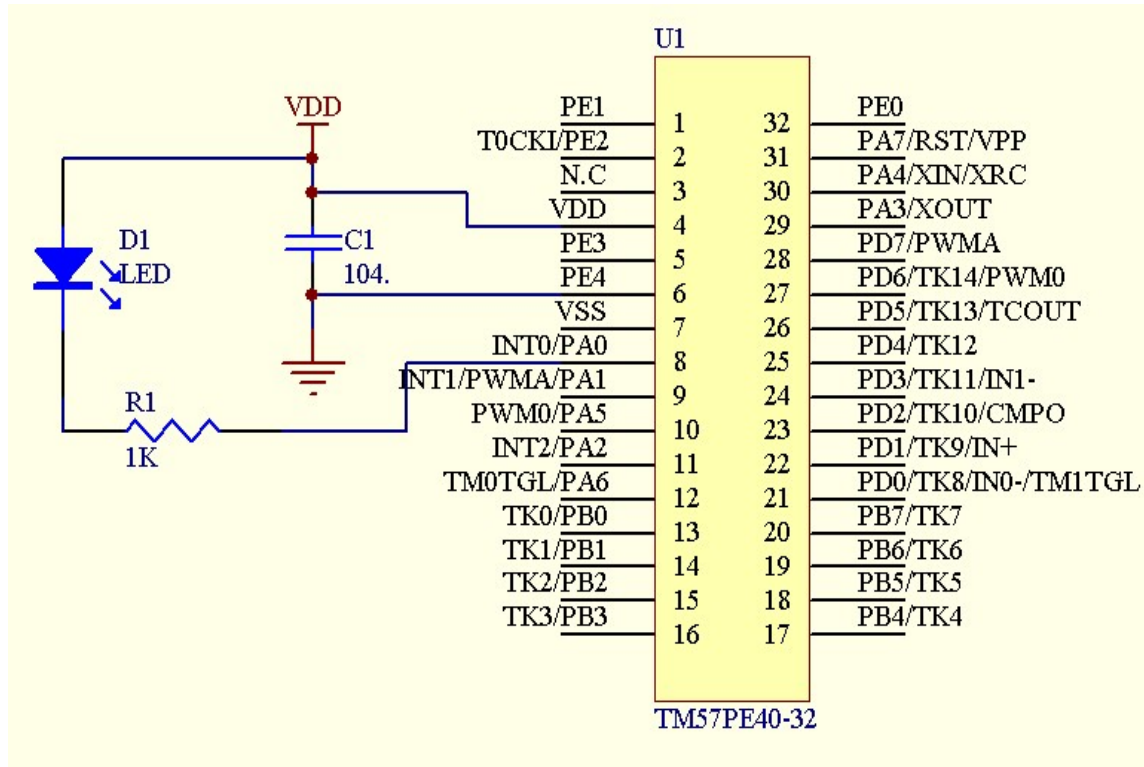
$1/4\text{M} * 2 * 16 * (256 - 131) = 1000 \text{ us}$ , the time can be used for oscilloscope to observe LED light level changes.

**3-3.** Timer0 configuration steps:

1. Set Timer0 clock source SELT0I and frequency division ratio TM0PSC.
2. Set Timer0 initial value.
3. Clear Timer0 interrupt flag tm0if=0 and enable interrupt tm0ie=1.



## 4. Circuit diagram



## 04. Timer1 Application Sample

**1. Sample function:** After power on, the LED output frequency is 500 Hz. Details of DEMO routine please refer to Timer1.asm.

### 2. Timer1 related registers

F-Plane

Address	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
08H	cmplie	Tm2ie	tm1i	tm0ie	wktie	xint2ie	xint1ie	xint0ie
09H	cmplif	Tm2if	Tm1if	Tm0if	wktif	Xint2i	Xint1i	Xint0i
0AH	Tm1l7	Tm1l6	Tm1l5	Tm1l4	Tm1l3	Tm1l2	Tm1l1	Tm1l0
0BH	Tm1h7	Tm1h6	Tm1h5	Tm1h4	Tm1h3	Tm1h2	Tm1h1	Tm1h0

R-Plane

Address	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0Ch	Tcopsc1	Tcopsc0	tcoe	-	Int1edge	Tm1oe	T1cap	Tm1psc

### 3. Sample description

**3-1.** Timer1 is 16-bit auto-load timer, when initial value is up to TM1L and TMH, when timer overflows, it is not necessary to set timer initial value.

**3-2.** Timer1 clock source can only be FOSC/2, the time calculation of Timer1 is as follows:

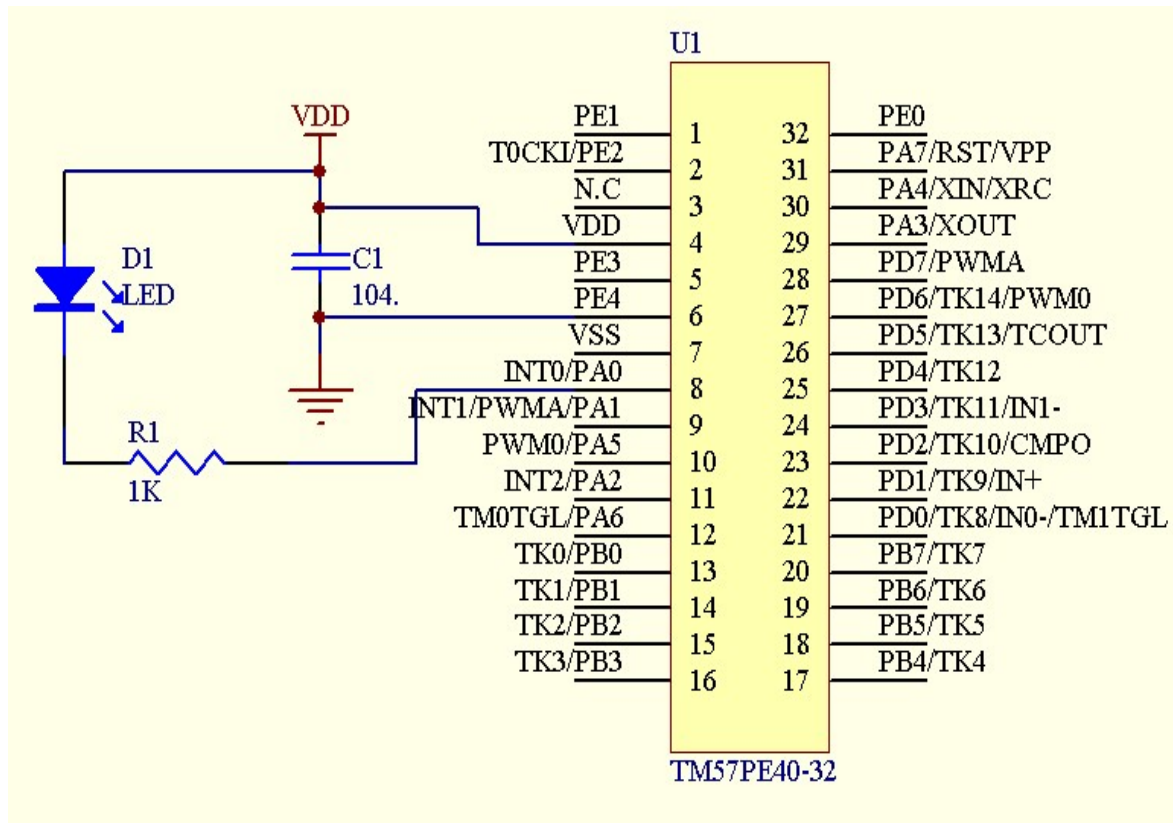
$$\text{Time} = 1/\text{fosc} * 2 * \text{Timer1 Pre-Scale} * (65536 - \text{TIMER1})$$

Use DEMO as sample, timer 1 ms calculation is as follows:  $1/4 * 2 * 16(256 - 131) = 1000 \text{ us}$ , the time can be used for oscilloscope to observe LED light level changes.

**3-3.** Timer1 configuration steps:

- (1) Set TIMER1 frequency division ratio, in this case, there are 2 divisions (TM1PSC=0, r\_plane, 0cH.0), in this case, it is set as timer mode (T1cap=0, r\_plane, 0cH.1).
- (2) Assign Timer1 initial value and reload value to TM1L (f\_plane, 0aH.7~0) and TM1H (f\_plane, 0bH. 7~0).
- (3) Clear Timer1 interrupt flag (tm1if=0, f\_plane, 09H.5) and enable interrupt (tm1ie=1, f\_plane, 08H.5).

## 4. Circuit diagram



## 05. TIMER2 Application Sample

1. Sample function: After power on, the LED output frequency is 1 Hz. Details of DEMO routine please refer to Timer2.asm.

### 2. Timer1 related registers

F-Plane

Address	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
08H	cmpie	Tm2ie	tm1i	tm0ie	wktie	xint2ie	xint1ie	xint0ie
09H	cmpif	Tm2if	Tm1if	Tm0if	wktif	Xint2i	Xint1i	Xint0i

R-Plane

Address	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0Eh	Wdtpsc1	Wdtpsc0	wdtslpstp	Tm2clks	Fircsel1	Fircsel0	Tm2psc1	Tm2psc0

### 3. Sample description

**3-1.** Timer2 is a 15-bit auto-load timer, the clock source can be Fcpuclk/128 or slow-clock, it is not necessary to set the initial value, user only needs to select the frequency division.

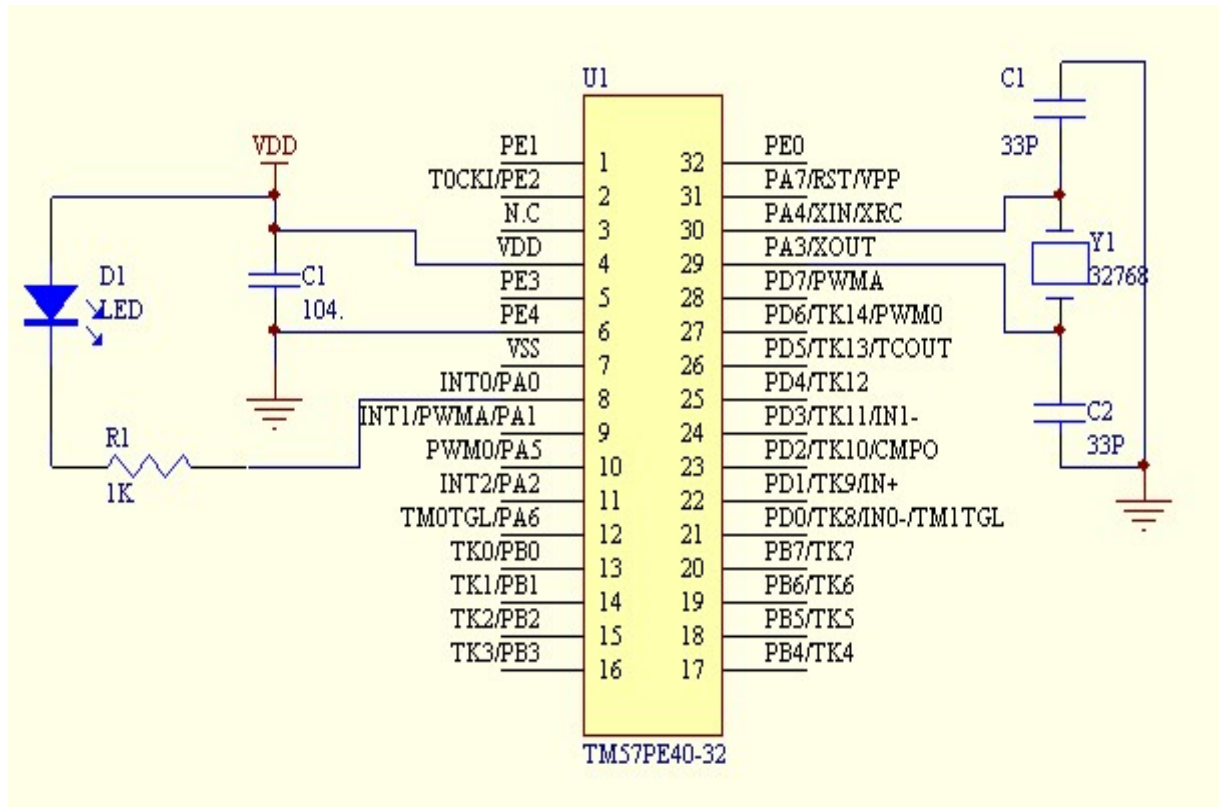
**3-2.** Set clock source and frequency division TM2PSC.

- (1) Select slow oscillating, timer 0.5 s:  $\text{Time} = 1/32768 * 16384 = 0.5\text{s}$
- (2) Select Fcpuclk/128, timer:  $\text{Time} = \text{Fcpuclk}/128 * 16384$

**3-3.** Timer2 configuration steps:

- (1) Set clock source and frequency division TM2PSC.
- (2) Clear Timer2 interrupt flag tm2if=0 and enable interrupt tm2ie=1.

## 4. Circuit diagram



## 06. WDT/WKT/XINT Application Sample

1. Sample function: under sleep mode, WKT and external interrupt is waken up from sleep mode. Details of DEMO please refer to WKT.asm, XINT.asm.

### 2. WDT/WKT/XINT related registers

#### F-Plane

Address	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
08H	cmpie	Tm2ie	tm1i	tm0ie	wktie	xint2ie	xint1ie	xint0ie
09H	cmpif	Tm2if	Tm1if	Tm0if	wktif	Xint2i	Xint1i	Xint0i
0FH	-	Sircsel1	Sircsel0	stpck	selsub	sube	Subtyp1	Subtyp0

#### R-Plane

Address	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
05H	Pae7	Pae6	Pae5	Pae4	Pae3	Pae2	pae1	pae0
08H	papu7	papu6	papu5	papu4	papu3	papu2	papu1	papu0
0BH	Pwm0psc1	Pwm0psc0	pwmae	Pwm0e	Pwm0inv	Tm0out	Wktpsc1	Wktpsc0
0CH	Tcopsc1	Tcopsc0	tcoe	-	Int1edge	Tm1oe	T1cap	Tm1psc
0DH	Wktkrcs	Tm0tks	Tkspeed1	Tkspeed0	Tksel3	Tksel2	Tksel1	Tksel0
0EH	Wdtpsc1	Wdtpsc0	wdtstp	Tm2clks	Fircsel1	Fircsel0	Tm2psc1	Tm2psc0

### 3. Sample description

**3-1.** When routine executing sleep, status register is PD=1, TO=0, it will enter power save mode. At this time, it can be waken up by reset (power on reset, low voltage reset, external pin (PA7) reset, watch dog timer reset) or external interrupt (3 external interrupts, WKT interrupt).

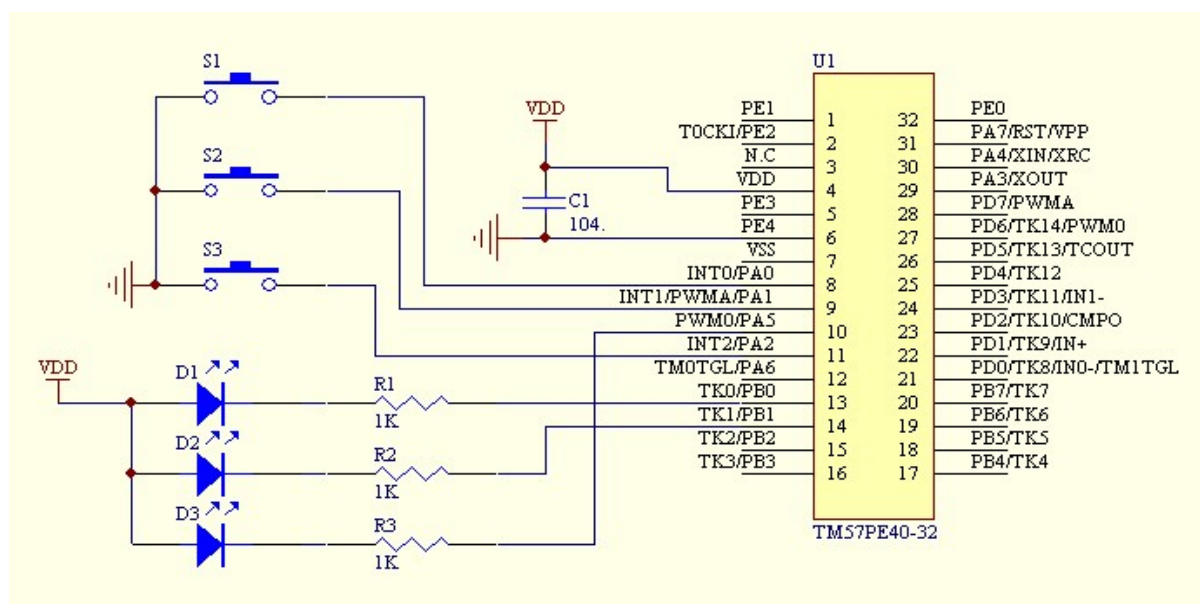
- (1) After power on reset, all system and external registers will reset to their default hardware setting.
- (2) Low voltage reset is set by SYSCFG Bit 11-10 LVR (LVR=11 is 1.5V, LVR=01 is 2.3V, LVR=00 is 3.2V).
- (3) External pin reset is set by SYSCFG Bit 7 XRESETE (1: enable, 0: disable).
- (4) WKT interrupt wake up from SLEEP configuration steps: (refer to WKT.asm)
  - System sets register SYSCFG, WDTE=0.
  - Set WDT overflow time, decided by WKTPSC.
  - Clear WKT interrupt flag wkti=0 and enable interrupt wktie=1, then execute SLEEP instruction to enter sleep mode.
  - WKT timer overflows, interrupt occurs, sleep mode is waken up.
- (5) External interrupt wake up from SLEEP configuration steps: (refer to XINT.asm)
  - INT0 and INT2 can be triggered only by falling edge, in application, turn on internal pull high resistor and set as input; INT1 is bidirectional triggered, decided by int1edge (0: falling edge, 1: rising edge). Set falling edge triggered, set as input port

and turn on internal pull high resistor. Set falling edge triggered, set as input port, and close internal pull high resistor and connect pull low resistor.

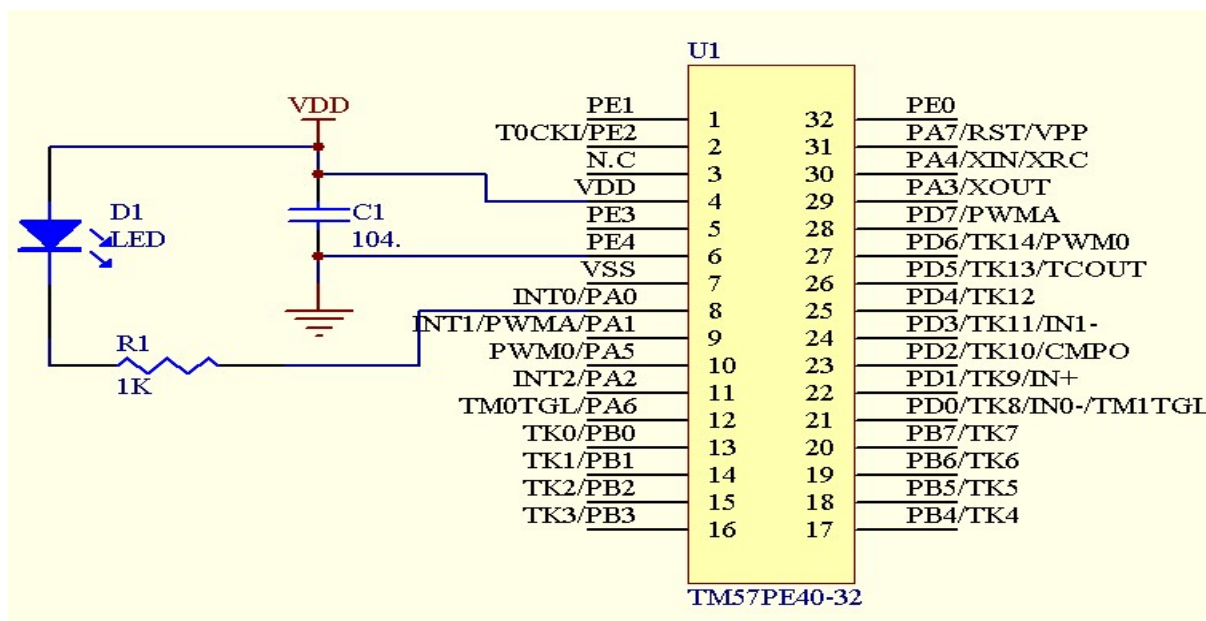
- Clear external interrupt flag  $xint2i=0$ ,  $xint1i=0$ ,  $xint0i=0$  and enable interrupt  $xint2e$ ,  $xint1e$ ,  $xint0e$ , execute SLEEP instruction to enter sleep mode.
- External interrupt pin changes from sleep mode to wake up, and enter external interrupt routine.

**3-2.** When IC is operating under general mode, WDT/WKT/XINT operating configuration steps are the same, except enter sleep mode.

#### 4. Circuit diagram



XINT circuit diagram



WKT circuit diagram

## 07. TK Application Sample

1. Sample function: TK7(PB7) controls LED light. Details of DEMO routine please refer to TK.asm.

### 2. TK related registers

#### F-Plane

Address	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
01H	Timer0 content							
06H	Pbd7	Pbd6	Pbd5	Pbd4	Pbd3	Pbd2	Pbd1	Pbd0
08H	cmple	Tm2ie	tm1i	tm0ie	wktie	xint2ie	xint1ie	xint0ie
09H	cmpif	Tm2if	Tm1if	Tm0if	wktif	Xint2i	Xint1i	Xint0i
0FH	-	Sircsel1	Sircsel0	stpfck	selsub	sube	Subtyp1	Subtyp0
12H	-	kicke	Clr pwm0	Clr tm2	Set tm1	Clr tm1	Stoptm1	Stoptm0

#### R-Plane

Address	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
02H	capola	T0cap	T0iedge	Selt0i	Tm0psc3	Tm0psc2	Tm0psc1	Tm0psc0
06H	Pbe7	Pbe6	Pbe5	Pbe4	Pbe3	Pbe2	pbe1	pbe0
07H	Pde7	Pde6	Pde5	Pde4	Pde3	Pde2	pde1	pdbe0
09H	pbpu7	pbpu6	pbpu5	pbpu4	pbpu3	pbpu2	pbpu1	pbpu0
0AH	pdpu7	pdpu6	pdpu5	pdpu4	pdpu3	pdpu2	pdpu1	pdpu0
0BH	Pwm0psc1	Pwm0psc0	pwmae	Pwm0e	Pwm0inv	Tm0out	Wktpsc1	Wktpsc0
0DH	Wktkrscs	Tm0tks	Tkspeed1	Tkspeed0	Tksel3	Tksel2	Tksel1	Tksel0
12H	Pb_ie7	Pb_ie6	Pb_ie5	Pb_ie4	Pb_ie3	Pb_ie2	Pb_ie1	Pb_ie0
13H	Pd_ie7	Pd_ie6	Pd_ie5	Pd_ie4	Pd_ie3	Pd_ie2	Pd_ie1	Pd_ie0

### 3. Sample description:

**3-1.** TM57PE40 has 15 touch keys. timer0, wkt, timer1 can be used as timer and counter.

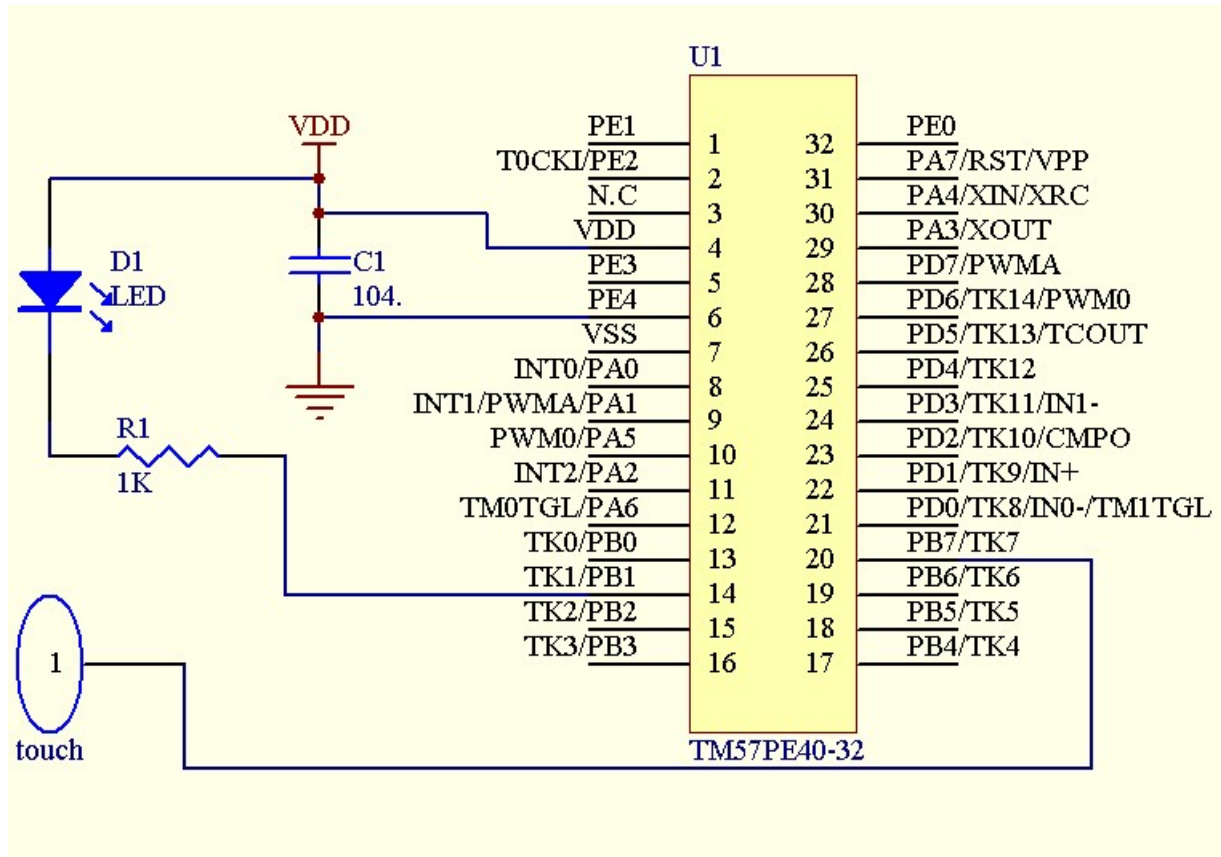
**3-2.** The variables available: touch key oscillating speed (tkspeed), channel selection (tksel), I/O selection (pb\_ie and pd\_ie), timer (tm0tks), counter (wktkrscs).

**3-3.** touch key configuration:

- (1) Close pull high, set pbpu, pdpu.
- (2) Enable touch key, set tkctl.
- (3) Enable I/O setting Pb\_ie, pd\_ie.



## 4. Circuit diagram:



## 08. TCOU Application Sample

1. Sample function: CPUCLK output PD5 PIN. Details of DEMO routine please refer to TCOU.asm.

### 2. TCOU related registers

F-Plane

Address	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0FH	-	Sircsel1	Sircsel0	stpfck	selsub	sube	Subtyp1	Subtyp0

R-Plane

Address	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0CH	Tcopsc1	Tcopsc0	tcoe	-	Int1edge	Tm1oe	T1cap	Tm1psc

### 3. Sample description:

**3-1.** TCOU will output frequency division CPUCLK to PD5.

**3-2.** Set TCOU:

- (1) In this sample, the fast clock is chosen as CPUCLK (selsub=0, f\_plane,0fH.3).
- (2) Cpuclk frequency division, in this case, is using 16 divisions (TCOPSC=11b, r\_plane,0cH.7~6).
- (3) Enable PD5 cpuclk output, set (TCOE=1, r\_plane,0cH.5)

### 4. Circuit diagram:

