

TM57 系列

TM57FA40&PA20&PA40 DEMO CODE FOR TM57FA40&PA20&PA40 BASIC FUNCTIONS 范例

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

Version	Date	Description
V1.0	June, 2011	New release

CONTENTS

AMENDMENT HISTORY	2
PRODUCT NAME	4
TITLE	4
APPLICATION NOTE	4
01. ADC 应用范例	5
02. PWM 应用范例	6
03. Timer0 应用范例	8
04. Timer1 应用范例	9
05. WDT/WKT/XINT 应用范例.....	10
06. Buzzer 应用范例	12

PRODUCT NAME

TM57 系列 IC

TITLE

TM57FA40 PA20/40_ADC 应用范例

TM57FA40 PA20/40_PWM 应用范例

TM57FA40 PA20/40_Timer0 应用范例

TM57FA40 PA20/40_Timer1 应用范例

TM57FA40 PA20/40_WKT/WDT/XINT 应用范例

TM57FA40 PA20/40_Buzzer 应用范例

APPLICATION NOTE

1. TM57PA20 与 TM57PA40 功能上是相同,区别在于 TM57PA20 OTP ROM 是 2K 不支持二次烧录功能, TM57PA40 OTP ROM 是 4K 支持二次烧录功能。

2. TM57PA40 与 TM57FA40 功能上基本相同, 细节上存在一定差异。如相互替换需请参考 TM57 系列 TM57PA40 与 TM57FA40 差别 **Application Note** ([AP-TM57PA40\(FA40\)_01SV10.pdf](#)) , 请登入 <http://www.tenx.com.tw> 下载相关资料。

01. ADC 应用范例

1. 范例功能：测试 **ADC0 (PA6)** 输入电压，具体 **DEMO** 程序请参 **ADC.asm** 。

2. ADC 相关寄存器

F-Plane

Address	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
05H	pad7	pad6	pad5	pad4	pad3	pad2	pad2	pad0
06H	pbd7	pbd6	pbd5	pbd4	pbd3	pbd2	pbd1	pbd0
07H	pdd7	pdd6	pdd5	pdd4	pdd3	pdd2	pdd1	pdd0
10H	adcdq11	adcdq10	adcdq9	adcdq8	adcdq7	adcdq6	adcdq5	adcdq4
11H	adcdq3	adcdq2	adcdq1	adcdq0	adcstart	adcse12	adcse11	adcse10

R-Plane

Address	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
05H	—	pae6	pae5	pae4	pae3	pae2	pae1	pae0
06H	—	—	—	—	—	—	pbe1	pbe0
07H	pde7	pde6	pde5	pde4	pde3	pde2	pde1	pde0
08H	—	npapu6	npapu5	npapu4	npapu3	npapu2	npapu1	npapu0
09H	—	—	—	—	—	—	npbpu1	npbpu0
0AH	npdpu7	npdpu6	npdpu5	npdpu4	npdpu3	npdpu2	npdpu1	npdpu0
0CH	buzen	adclks2	adclks1	adclks0	tm1psc3	tm1psc2	tpm1psc1	tmp1psc0
12H	nadcie7	nadcie6	nadcie5	nadcie4	nadcie3	nadcie2	nadcie1	nadcie0

3. 范例说明

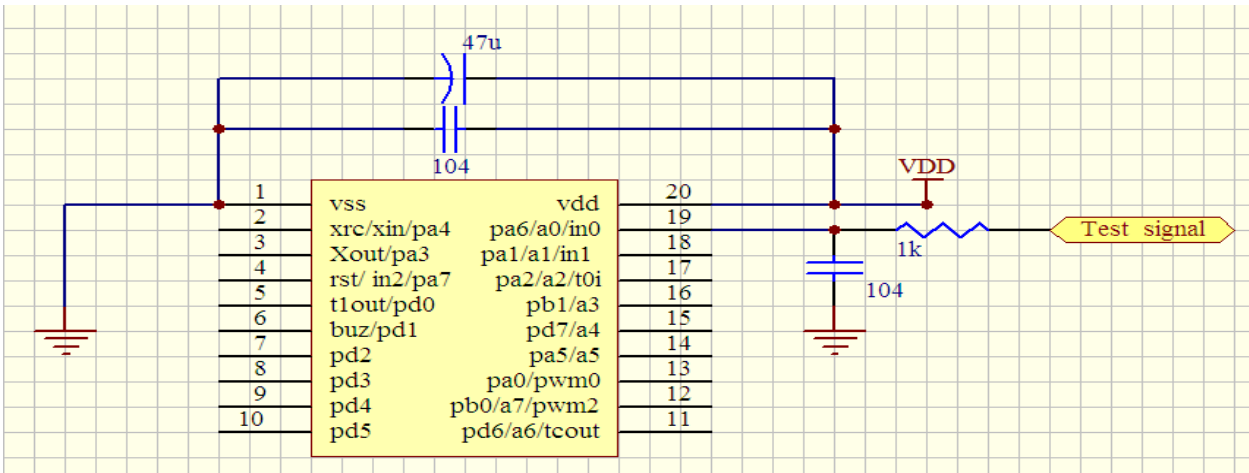
3-1. TM57FA40 PA20/40 是 12-bit 模数转换器，参考电压 VDD。应用时 AD 口接一个 R,C 组成低通滤波器，其作用是保护 AD 口和保证 ADC 精度,一般 R 取 1 KΩ,C 取 104p。

3-2. AD 转换步骤：

- (1) 将转换 AD 口设置为输入口关闭上拉，对应 I/O 端口数据寄存器置 1。如本例选 ADC0 设置为：pae6=0(R_plane,05.6);npapu6=1(R_plane,08.6); pad6=1(F_plane,05.6)。
- (2) 选择 ADC 的转换通道。ADCSEL=0(F_plane,11h.2-0)。
- (3) 将所须 I/O 设置为 ADC 类型。nadcie0=0(R_plane,12h.0)。
- (4) 设置 ADC 转换时钟频率（选择适当 ADC 转换频率,最高不可以超过 2 MHz），一次 AD 转换大约需要 50 个 AD 时钟周期。ADCLKS=1(R_plane,0ch.6-4)。
- (5) 打开 ADC 转换。ADCSTART=1(F_plane,11h.3)。
- (6) 等待 ADC 转换完成（当 ADC 转换完成时,硬件自动将 ADCSTART 清零）。

(7) ADC 数据处理。ADC 结果存放于 ADCDQ(F_plane,10h.7-0;F_plane,11h.7-4)。ADC 每次转换完成时，需要将结果存储到指定的位置，否则下一次转换的结果会覆盖 ADC 数据寄存器内容。

4. 线路图



02. PWM 应用范例

1. 范例功能：上电后 PA0 输出周期为 64 us,占空比为 1/2 波形，PB0 输出周期为 64 us,占空比为 1/4 波形，具体 DEMO 程序请参考 PWM.asm。

2. PWMA/B 相关寄存器

F-Plane

Address	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0CH	pwm0duty9	pwm0duty8	pwm0duty7	pwm0duty6	pwm0duty5	pwm0duty4	pwm0duty3	pwm0duty2
0DH	pwm0duty1	pwm0duty0						
0EH	pwm1duty9	pwm1duty8	pwm1duty7	pwm1duty6	pwm1duty5	pwm1duty4	pwm1duty3	pwm1duty2
0FH	pwm1duty1	pwm1duty0						

R-Plane

Address	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0BH	buzout	pwm0out	pwm1out	int1edge	tcout	tm1out	wktpsc1	wktpsc0

3. 范例说明

3-1. PWMA/PWMB 是有类似结构，其时钟源是系统时钟 F_{osc} 。当 8 位计数器的值和 PWM 占空比寄存器(PWMDUTY)的高 8 位 MSB 相等时，PWM 输出信号将变为低电平,低 2 位 LSB 决定 PWM 的输出信号是立即变高还是延迟一个时钟周期后才变为高。

以 4 个 PWM 周期为例：

PWMDUTY[1:0]=0，每个周期溢出后都是延时一个时钟周期才变高；PWMDUTY[1:0]=1，只有第三个周期溢出后直接变高；

PWMDUTY[1:0]=2，只有第二、四个周期溢出后直接变高；

PWMDUTY[1:0]=3，只有第二、三、四个周期溢出后直接变高。

即高电平分别增加了 0，1，2，3 时钟周期，由此可以将 PWM 的 duty 解析度扩展为 1024 个时钟周期。

3-2. PWM 输出设置步骤：

- (1) 将 PA0、PB0 口设置为输出，并且关闭上拉。
`pae0=1(R_plane,05.0);npapu0=1(R_plane,08.0);`
`pbe0=1(R_plane,06.0);npbpu0=1(R_plane,09.0)。`
- (2) 设置 PWMA/B 的 duty 值。PWM0DUTY(F_plane,0ch.7-0; F_plane,0dh.7-6),PWM1DUTY(F_plane,0eh.7-0; F_plane,0fh.7-6)。
- (3) 使能 PWMA/B 输出。pwm0out=1(R_plane,0bh.6),pwm1out=1(R_plane,0bh.5)。
- (4) 关闭 PWMA/B 输出。pwm0out=0(R_plane,0bh.6),pwm1out=0(R_plane,0bh.5)。

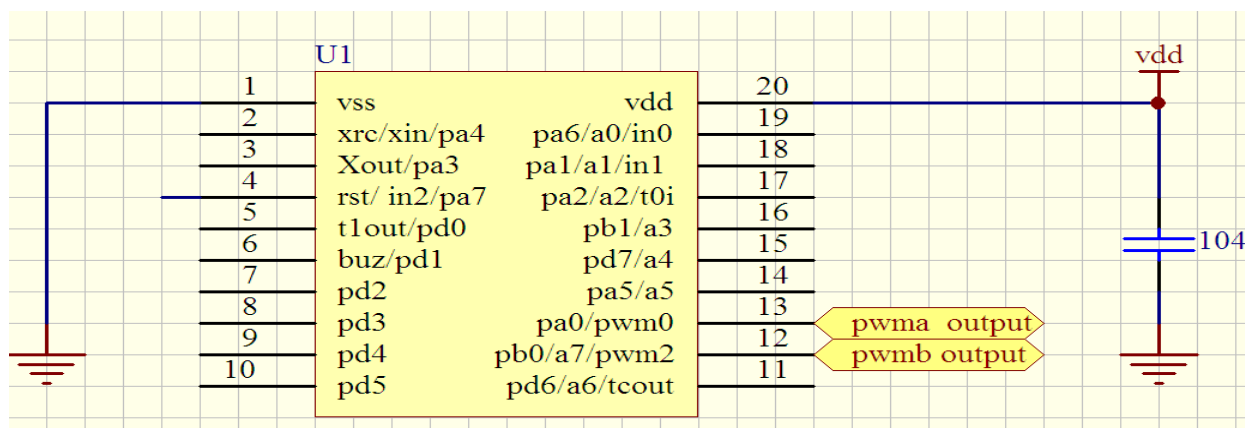
3-3. 周期计算公式：周期 $T=(1/f_{osc}) \times 256$

以程序设置值为例： $1/4M \times 256 = 0.25 \mu s \times 256 = 64 \mu s$

3-4. 占空比计算公式：占空比 $duty = PWMDUTYH/256 + PWMDUTYL/1024$

例如：PWMDUTYH=128, PWMDUTYL=2, 则 $duty = 128/256 + 2/1024 = 514/1024$

4. 线路图



03. Timer0 应用范例

1. 范例功能：上电后 LED 输出频率是 500 Hz。具体 DEMO 程序请参考 Timer0.asm

2. Timer0 相关寄存器

F-Plane

Address	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
01H	Timer0 content							
09H	—	—	tm1i	tm0i	wkti	xint2i	xint1i	xint0i

R-Plane

Address	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
02H	tcoutpsc1	tcoutpsc0	t0iedge	selt0i	tm0psc3	tm0psc2	tm0psc1	tm0psc0
0EH	—	—	tm1ie	tm0ie	wktie	xint2e	xint1e	xint0e

3. 范例说明

3-1. 定时器 0 是 8-bit 递增定时计数器，当定时器计数从 0xff 到 0 时溢出发生定时中断。不具有重载功能，所以中断后需要重新设置 TIMER0 的初值。TIMER0 时钟源选择由位 SELT0I 决定，SELT0I=0 时钟源是 fosc/2，SELT0I 时钟源从 T0I pin 输入。

3-2. 定时时间计算公式：

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

外部计数计算公式：

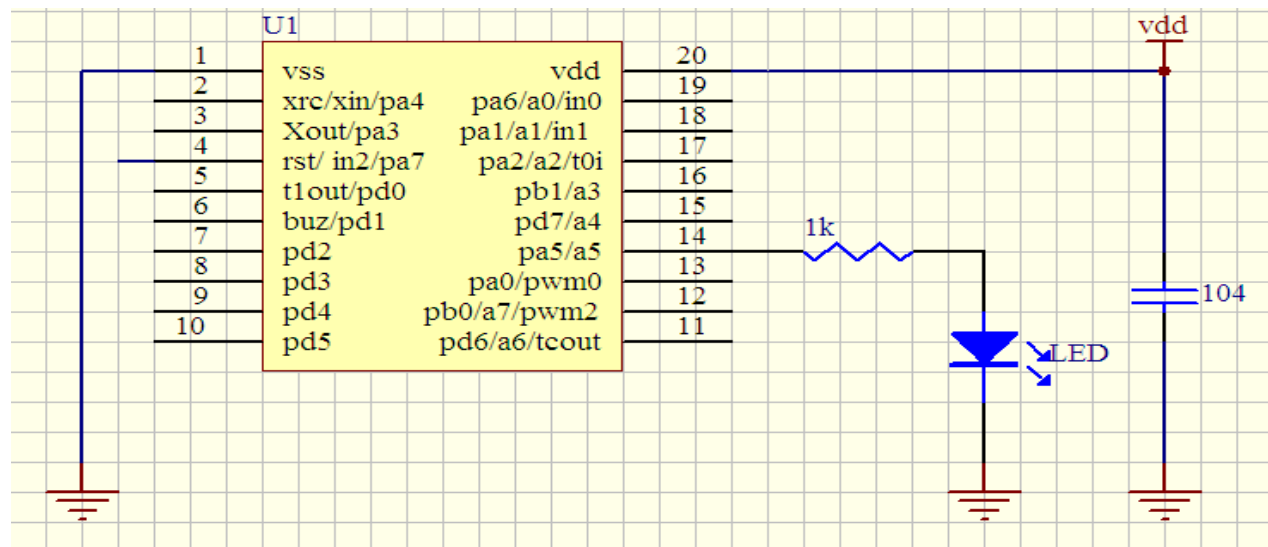
$$\text{Tcount} = 1/\text{Ftoi} * \text{Timer0 Pre-Scale} * (256 - \text{TIMER0}) \quad (\text{Ftoi 为 Toi 输入频率})$$

以 DEMO 为例，定时 1 ms 计算如下： $1/4\text{M} * 2 * 16 * (256 - 131) = 1000 \text{ us}$ ，时间值可用示波器观察 LED 灯电平变化。

3-3. Timer0 设置步骤：

- (1) 设置 TIMER0 的时钟来源 SELT0I(R_plane,02.4)和分频比 TM0PSC(R_plane,02.3-0)。
- (2) 设置定时器 0 的初始值。timer0(F_plane,01.7-0)。
- (3) 清定时器 0 的中断标志 tm0i=0(F_plane,09.4)和使能中断 tm0ie=1(R_plane,0eh.4)。

4. 线路图



04. Timer1 应用范例

1. 范例功能：上电后 LED 输出频率 500 Hz。具体 DEMO 程序请参考 Timer1.asm
2. Timer1 相关寄存器

F-Plane

Address	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0AH	Timer1 content							
09H	—	—	tm1i	tm0i	wkti	xint2i	xint1i	xint0i

R-Plane

Address	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0CH	buzen	adclks2	adclks1	adclks0	tm1psc3	tm1psc2	tpm1psc1	tpm1psc0
0EH	—	—	tm1ie	tm0ie	wktie	xint2e	xint1e	xint0e
0DH	Timer1 reload offset value while it rolls over							

3. 范例说明

- 3-1. 定时器1是8bit自动加载定时器，当给定初值到timer1(F_plane,0ah.7-0)和tm1reload (R_plane,0dh.7-0)后，定时器溢出后不需重新给定时器赋初值。

3-2. 定时器 1 的时钟源只能来自 FOSC/2，定时器 1 的时间计算如下：

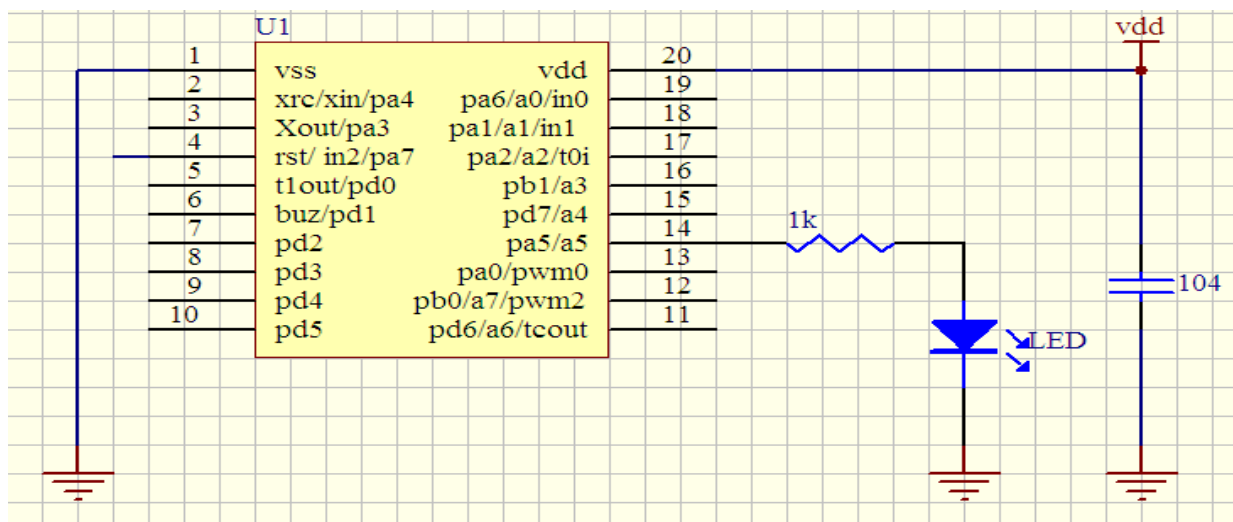
$$\text{Time} = 1/\text{fosc} * 2 * \text{Timer1 Pre-Scale} * (256 - \text{TIMER0})$$

以 DEMO 为例，定时 1ms 计算如下： $1/4 * 2 * 16(256 - 131) = 1000 \text{ us}$ ，时间值可用示波器观察 LED 灯电平变化。

3-3. 定时器 1 设置步骤：

- (1) 设置 TIMER1 的分频比。TM1PSC(R_plane,0ch.3-0)。
- (2) 把定时器 1 的初始值送给 timer1(F_plane,0ah.7-0)和 tm1reload(R_plane,0dh.7-0)。
- (3) 清定时器 1 的中断标志位 tm1i=0(F_plane,09.5)和使能中断位 tm1ie=1(R_palen,0eh.5)。

4. 线路图



05. WDT/WKT/XINT 应用范例

1. 范例功能：在睡眠模式下，WKT 和外部中断从睡眠模式中唤醒，具体 DEMO 程序请参考 WKT.asm,XINT.asm

2. WDT/WKT/XINT 相关寄存器

F-Plane

Address	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
03H	—	—	rambank	to	pd	z	dc	c
09H	—	—	tm1i	tm0i	wkti	xint2i	xint1i	xint0i

R-Plane

Address	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0BH	buzout	pwm0out	pwm1out	int1edge	tcout	tm1out	wktpsc1	wktpsc0
0EH	—	—	tm1ie	tm0ie	wktie	xint2e	xint1e	xint0e

3. 范例说明

3-1. 当程序执行 **sleep** 时，状态寄存器位 **PD=1**，**TO=0**，程序即进入省电模式，此时可因复位（上电复位、低压复位、外部引脚（**PA7**）复位、看门狗复位）或外部中断（三个外部中断 **WKT** 中断）而唤醒。以下分别介绍：

- （1）上电复位后所有系统和外围寄存器都恢复到它们默认的硬件值。
- （2）低压复位由 **SYSCFG** 的 **Bit11-10 LVR** 设定 (**LVR=11** 是 2.1V，**=01** 是 3.1V)。
- （3）外部引脚复位由 **SYSCFG** 的 **Bit7 XRESETE** 设定（**=1** 允许，**=0** 禁止）。
- （4）看门狗复位由 **SYSCFG** 的 **Bit6 WDTE** 设定(**WDTE=0** 时看门狗复位禁止，**WKT** 定时器启用；**WDTE=1** 时看门狗复位启用，**WKT** 定时器禁止)。 **WKT** 和 **WDT** 是同一个内部 **RC** 定时器，在普通模式和睡眠模式下运行情况由 **WDTE** 和 **WKTIE** 决定，如下表：
WKTIE(R_plane,0eh.3)为 **WKT** 中断使能位。

模式	WDTE	WKTIE	片内RC振荡
普通模式	0	0	停止工作
	0	1	运行
	1	0	运行
	1	1	运行
省电模式	0	0	停止
	0	1	运行
	1	0	停止
	1	1	运行

（5）**WDT** 复位从 **SLEEP** 唤醒设置步骤：（应用时参考 **WDT.asm**）

- 系统配置寄存器 **SYSCFG** 的位 **WDTE=1**。
- 设定 **WDT** 溢出时间，由 **WKTPSC(R_plane,0bh.1-0)**决定。
- 使能位 **wktie=1(R_plane,0eh.3)**并执行 **SLEEP** 指令进入睡眠模式。
- 看门狗定时器溢出复位从睡眠模式唤醒。

（6）**WKT** 中断从 **SLEEP** 唤醒设置步骤：（应用时参考 **WKT.asm**）

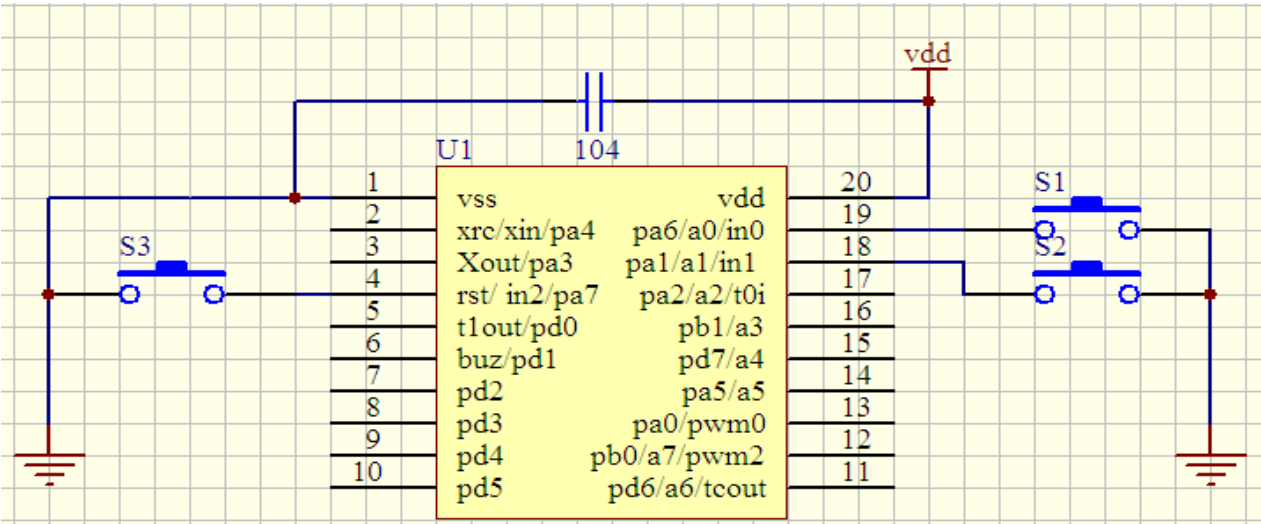
- 系统配置寄存器 **SYSCFG** 的位 **WDTE=0**。
- 设定 **WDT** 溢出时间，由 **WKTPSC (R_plane,0bh.1-0)**决定。
- 清 **WKT** 中断标志 **wkti=0(F_plane,09.3)**和使能中断 **wktie=1(R_palen,0eh.3)**后执行 **SLEEP** 指令进入睡眠模式。
- 看门狗定时器溢出进入中断，睡眠模式被唤醒。

(7) 外部中断 SLEEP 唤醒设置步骤：（应用时参考 XINT.asm）

- INT0 和 INT2 只能下降沿触发，应用时打开内部上拉电阻并设为输入口；INT1 为双向触发，由位 `int1edge(R_plane,0b.4)` 决定（=0 下降沿，=1 上升沿）。设下降沿触发，设为输入口并打开内部上拉电阻；设下降沿触发，设为输入口、关闭内部上拉电阻并外接下拉电阻。
- 清外部中断标志位 `xint2i=0,xint1i=0,xint0i=0(F_plane,09.2-0)` 和使能中断位 `xint2e,xint1e,xint0e(R_palen,0eh.2-0)`，执行 SLEEP 指令进入睡眠模式。
- 外部中断引脚变化从睡眠模式中唤醒并进入外部中断程序。

3-2. IC 工作在普通模式时，WDT/WKT/XINT 工作设置步骤一样,不进入睡眠模式即可。

4. 线路图



06. Buzzer 应用范例

1. 范例功能：PD1 脚输出频率 3.2K，具体 DEMO 程序请参考 Buzzer.asm

2. Buzzer 相关寄存器

R-Plane

Address	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0BH	buzout	pwm0out	pwm1out	int1edge	tcout	tm1out	wktpsc1	wktpsc0
0CH	buzen	adclks2	adclks1	adclks0	tm1psc3	tm1psc2	tpm1psc1	tmp1psc0
10H	buzpsc1	buzpsc0	buzprod5	buzprod4	buzprod3	buzprod2	buzprod1	buzprod0

3. 范例说明

3-1. 蜂鸣驱动器包含 6Bit 计数器和时钟分频器，产生 50%占空比方波。设定周期和分频后使能 BUZOUT(R_plane,0bh.7)，BUZEN(R_plane,0ch.7)即可输出设定的频率。

3-2. Buzzer Counter 每溢上一次 Buzzer output 即取反一次，所以输出频率为其一半。

3-3. 当关闭蜂鸣输出时在蜂鸣计数器下一个周期到来时才能关闭，同样当蜂鸣器计数值改变时也在蜂鸣计数器下一个周期到来时才能改变输出频率。

3-4. 输出频率计算公式

$$F_{bz} = (f_{osc}/2)/(Instruction\ Cycle\ Divider)/(BUZ_PROD + 1)/2$$

3-5. Buzzer 设置步骤: (应用时参考 Buzzer.asm)

- (1) 把 Buzzer 设为输出关掉上拉。pde1=1(R_plane,07.1);npdpu1=1(R_plane,0ah.1)。
- (2) 使能蜂鸣输出到 PD1 脚，即 buzout=1(R_plane,0bh.7)。
- (3) 选择蜂鸣时钟分频 BUZ_PSC(R_plane,10h.7-6)和蜂鸣周期值 BUZ_PROD(R_plane,10h.5-0)。
- (4) 允许蜂鸣计数器输出，即 buzen=1(R_plane,0ch.7)。

4. 线路图

