

TMS320F2833x 的时钟与系统控制

5.1 振荡器 OSC 和锁相环 PLL 模块

为了让 DSP 能够按部就班地执行相应的代码来实现相应的功能,各个外设、数据总线、程序总线相互之间密切配合,除了要给 DSP 提供电源外,还需要向 CPU 不断地提供规律的时钟脉冲信号,从而实现各个模块行为在时间上的紧密契合,而器件的时钟脉冲信号是由片上振荡器 OSC 和锁相环模块 PLL 共同决定的。因此,振荡器 OSC(Oscillator)、锁相环 PLL(Phase Locked Logic)等构成的系统及对应的时钟机制是 DSP 的“心脏”,为其提供正常运行的动力和节奏。

1. 振荡器 OSC 的基本介绍

振荡器的种类很多,按振荡激励方式有自激振荡器、他激振荡器;按电路结构有阻容振荡器、电感电容振荡器、晶体振荡器、音叉振荡器等;按输出波形有正弦波、方波、锯齿波等。振荡器是用来产生重复电子信号(通常是正弦波或方波)的电子元件,它可将直流电转换为具有一定频率交流电信号输出的电子电路或装置。

最常见的振荡电路是由电容器和电感器组成的 LC 回路,通过电场能和磁场能的相互转换产生自由振荡,为了维持振荡,还要有具有正反馈的放大电路。

2. 锁相环 PLL 的基本介绍

目前 DSP 集成芯片上的锁相环主要是通过软件实时地配置片上外设时钟,由于采用软件可编程锁相环,所设计的处理器外部允许较低的工作频率,而片内经过锁相环模块提供较高的系统时钟,这种设计可以有效地降低系统对外部时钟的依赖和电磁干扰,提高系统启动和运行时的可靠性,降低系统对硬件的设计要求。图 5-1 为锁相环方框图,由图可以看出,锁相环由鉴相器、低通滤波器和压控振荡器组成。

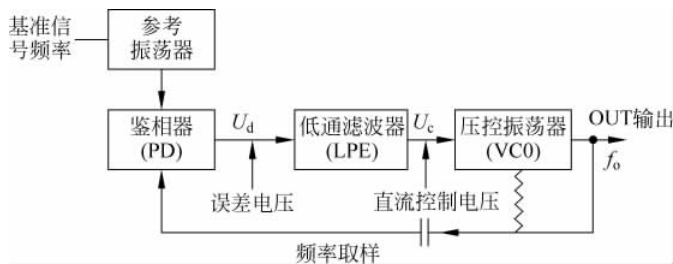


图 5-1 锁相环方框图

鉴相器用来鉴别输入信号 U_i 与输出信号 U_o 之间的相位差,并输出误差电压 U_d 。 U_d 中的噪声和干扰成分被低通性质的环路滤波器滤除,形成压控振荡器的控制电压 U_c 。当输出振荡频率 f_o 与输入信号频率 f_i 相等时,环路被锁定,即相位锁定。相位锁定时,压控振荡使输出信号跟随输入信号的变化。

3. 时钟信号形成

如图 5-2 所示,F2833x 可以通过外部振荡器或连接在芯片上的晶振电路提供时钟脉冲信号,并通过内部的 PLL 锁相环电路倍频后提供给 CPU。

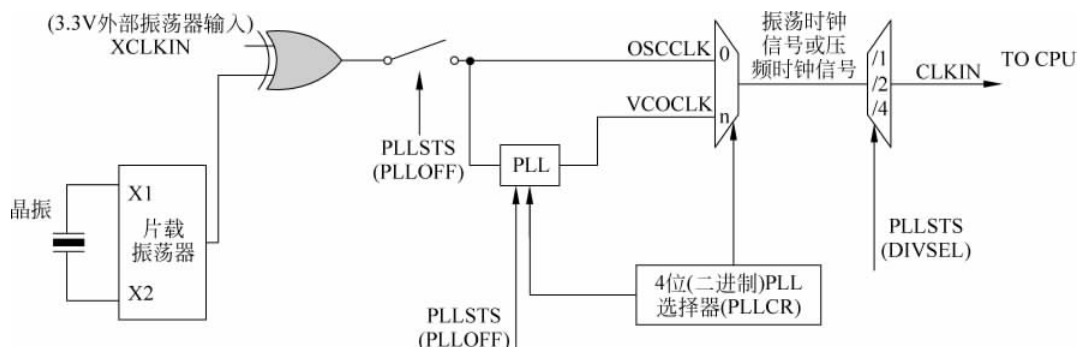


图 5-2 CPU 时钟信号生成框图

产生 OSCCLK 时钟信号的两种方式具体表述如下。

- (1) 晶体振荡器工作模式：外部晶体振荡器接在引脚 X1 和 X2 之间,为芯片提供时钟基准。
- (2) 外部时钟源工作模式：如果片载振荡器未被使用,那么振荡器可以使用下面配置的任何一個：

- XCLKIN 引脚接 3.3V 外部振荡器,X1 引脚在低电平,X2 引脚悬空；
- X1 引脚接 3.3V 外部振荡器,XCLKIN 引脚在低电平,X2 引脚悬空。

4. PLL 的 3 种配置模式

产生的 OSCCLK 时钟信号要交由 PLL 模块进行倍频控制,PLL 模块具有 3 种工作模式,具体如表 5-1 所示。

表 5-1 外部时钟信号与送至 CPU 时钟信号的关系

PLL 模式	描 述	PLLSTS[DIVSEL]	SYSCCLKOUT
PLL 被禁止	用户在 PLLSTS 寄存器中设置 PLLOFF 位可以使锁存环模块在该模式下被禁止。主要用于减少系统噪声和低功耗操作。在进入这种模式之前,PLLCCR 寄存器必须被清零(PLL 被旁路),输入到 CPU 的时钟信号直接来自 X1/X2,X1 或 XCLKIN	0,1	OSCCLK/4
		2	OSCCLK/2
		3	OSCCLK/1
PLL 被旁路	旁路是上电或外部复位后的默认配置。当 PLLCCR 寄存器必须被清零或被修改使得 PLL 锁住一个新的频率时,采用该模式。在这种模式下,锁相环自身被旁路,但 PLL 没有被关断	0,1	OSCCLK/4
		2	OSCCLK/2
		3	OSCCLK/1
PLL 被使能	通过向 PLLCCR 寄存器写入一个非零值来实现,直到 PLL 被锁存住,芯片才转换为旁路模式	0,1	OSCCLK/4
		2	OSCCLK/2

5.2 外设时钟信号

时钟信号产生后送往 CPU,通过 CPU 分配到各个外设,达到对外设动作行为的时间设定控制。DSP 整个系统以及它的各个外设模块从本质上来讲就是一个相对比较复杂的时序逻辑处理电路,各个外设模块的正常工作就需要对应的时钟信号。

使用这种时钟制的模型有以下 3 个优势。

- (1) 统一性: 各个模块间有一个相对统一的标准,便于相互之间的通信;
- (2) 协调性: 一个系统要有效地工作,必须依赖于各个模块之间的有机配合,而集中制为各个模块之间的有效配合提供了可能性;
- (3) 高效性: 集中制有利于资源的合理配置,是一个相对高效的系统。

几乎每个外设(如 SCI、EV、ADC 等)的正常工作都需要相应的时钟信号,各时钟信号都是对系统时钟信号 SYSCLKOUT 处理后产生的。在系统初始化的时候,需要对使用到的各个外设的时钟进行使能,与时钟使能相关的寄存器是 PCLKCR(Peripheral CLocKContRol),例如,某个程序里用到了外设 EVA、SCIA 和 ADC 这 3 个外设,则按照下面的程序对这 3 个外设进行时钟的使能。

```
SysCtrlRegs.PCLKCR.bit.SCIENCLKA=1;    //使能外设 SCIA 的时钟
SysCtrlRegs.PCLKCR.bit.EVAENCLK=1;      //使能外设 EVA 的时钟
SysCtrlRegs.PCLKCR.bit.ADCENCLK=1;      //使能外设 ADC 的时钟
```

SysCtrlRegs 由 System、Control、Register 这 3 个单词的缩写构成,PCLKCR 为与时钟使能相关的寄存器,bit 意味着操作对象是该寄存器对应的某一位,ADCENCLK 中 ADC 对应的是外设名称,而 EN 则对应使能控制功能,CLK 对应的是系统时钟信号,那么 ADCENCLK 所代表的含义也就不言而喻了。图 5-3 为外设时钟信号产生的整体框图。

从 CPU 发出的系统时钟信号 SYSCLKOUT 并不直接发送给各个外设,而是经过高低速外设时钟预定标寄存器,然后再分配到各个外设对各个外设进行控制。SYSCLKOUT 信号经过低速外设时钟预定标寄存器 LOSPCP 输出 LSPCLK,提供给低速外设 SCIA、SCIB、SPI 和 McBSP; SYSCLKOUT 信号经过高速外设时钟预定标寄存器 HISPCP 输出 HSPCLK,提供给高速外设 EVA、EVB 和 ADC。我们知道高低速时钟预定标寄存器的取值范围都是 0~7,而且二者的取值完全是相互独立的。这样会出现 LSPCLK 时钟信号频率高于 HSPCLK 时钟信号频率的情况(给 LOSPCP 寄存器所赋的值小于给 HISPCP 寄存器所赋的值)理论上的存在,这完全取决于用户对于外设时钟高低速预定标寄存器的初始化设定。但是一般情况下不会出现这样的情况,因为低速外设所需要的时钟要比高速外设所需要的时钟来得慢些。

实际上,各个外设在使用过程中,LSPCLK 或者 HSPCLK 还需要经过各个外设自己的时钟预定标,如果外设自己的时钟预定标的值为 0 的话,则外设实际使用的时候就是 LSPCLK 或者 HSPCLK。在实际使用时,为了降低系统功耗,不使用的外设最好将其时钟禁止。

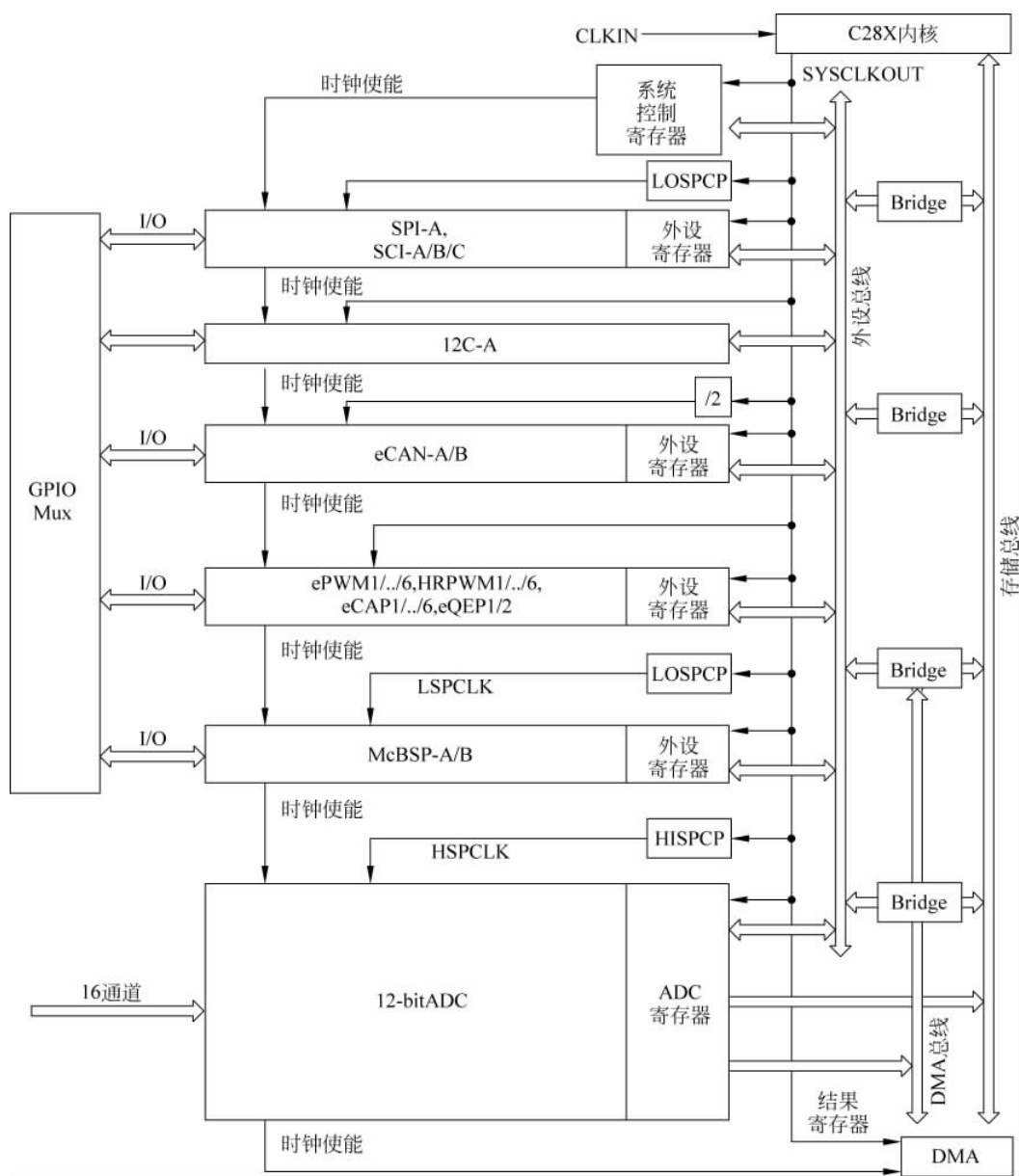


图 5-3 时钟信号系统

5.3 低功耗模式

F2833x 系列的 DSP 具有 3 种低功耗模式,每种工作模式的状态如表 5-2 所示。

表 5-2 低功耗模式

低功耗模式	LPMCR0[1:0]	OSCCLK	CLKIN	SYSCCLKOUT	退出方式
IDLE	00	On	On	On	看门狗中断任何使能中断
STANDBY	01	On(看门狗仍然运行)	Off	Off	看门狗中断,GPIO PortA 仿真器信号
HALT	1x	Off(振荡器及 PLL 停止工作,看门狗也停止工作)	Off	Off	GPIO PortA 仿真器信号

对这 3 种模式的具体描述如下。

1. IDLE 模式

在此模式下,CPU 可以通过使能中断或 NMI 中断退出该模式,LPM 单元将不执行任何工作。

2. STANDBY 模式

如果低功耗模式控制寄存器 LPMCR0 寄存器的 LPM 位设置为 01,当 IDLE 指令执行时,设备进入 STANDBY 模式。在该模式下,CPU 的输入时钟关闭,这使得所有来自 SYSCCLKOUT 的时钟都被关闭,振荡器和看门狗将一直起作用。

进入 STANDBY 模式之前,需要完成以下任务:

- (1) 在 PIE 模块中使能 WAKEINT 中断,该中断连接着看门狗和低功耗模式模块中断;
- (2) 在 GPIOLPMSE 寄存器中指定一个 GPIO 端口 A 信号唤醒设备,GPIOLPMSE 寄存器是 GPIO 模块的一部分。另外,使能 LPMCR0 寄存器,被选中的 GPIO 信号、 $\overline{\text{XRS}}$ 输入信号和看门狗中断信号可以将处理器从 STANDBY 模式中唤醒。

3. HALT 模式

如果低功耗模式控制寄存器 LPMCR0 寄存器的 LPM 位被设置为 1x,当 IDLE 指令被执行时,设备进入 HALT 模式。在该模式下,所有的设备(包括 PLL 和振荡器)均被锁住。进入 HALT 模式之前,需要完成以下任务:

- (1) 在 PIE 模块中使能 WAKEINT 中断(PIEIER1.8=1),该中断连接着看门狗和低功耗模式模块中断;
- (2) 在 GPIOLPMSE 寄存器中指定一个 GPIO 端口 A 信号唤醒设备,GPIOLPMSE 寄存器是 GPIO 模块的一部分。另外,被选中的 GPIO 信号、 $\overline{\text{XRS}}$ 输入信号可以将处理器从 HALT 模式中唤醒;
- (3) 尽可能地禁止除 HALT 模式唤醒中断之外的所有中断,在设备离开 HALT 模式后中断可以重新被使能;
- (4) 设备退出 HALT 模式所需要的条件是:PIEIER1 寄存器的第 7 位(INT1.8)必须是 1,IER 寄存器的第 0 位(INT1.8)必须是 1;
- (5) 如果以上条件都具备,那么:①如果 INTM=0,WAKE_INT ISR 首先被执行,之后是 IDLE 指令;②如果 INTM=1,WAKE_INT ISR 不被执行,IDLE 指令被执行。

当设备工作在 limp 模式($PLLSTS[MCLKSTS]=1$)时,不要进入 HALT 低功耗模式,否则,设备可能进入 STANDBY 模式或者出现死机而无法退出 HALT 模式。因此,在进入 HALT 模式之前,一直要检查 $PLLSTS[MCLKSTS]$ 位是否等于 0。

5.4 看门狗模块

看门狗电路的应用,使得 DSP 可以在无人监控的状态下实现连续工作。其工作原理是:看门狗电路和 DSP 的一个 I/O 引脚相连,该 I/O 引脚通过程序控制它定时地往看门狗的这个引脚上送入高电平(或是低电平),这一程序语句分散地放在 DSP 程序的其他控制语句中间。一旦由于不可知因素形成的干扰,导致 DSP 程序跑飞继而陷入某一程序、进入死循环状态时,写看门狗引脚的程序便不能被执行。这个时候,看门狗电路就会由于得不到 DSP 送来的信号,便在它与单片机复位引脚相连的引脚上送出一个复位信号,使得单片机发生复位,即程序从程序存储器的起始位置开始执行,这样便实现了 DSP 的自动复位。

1. 防止 WDCNTR 溢出

从图 5-4 可以看到,DSP 芯片 28335 的看门狗电路有一个 8 位看门狗加法计数器 WDCNTR,如果 WDCNTR 计数达到最大值,看门狗模块就会产生一个输出脉冲,脉冲的宽度为 512 个振荡器时钟宽度。为了防止 WDCNTR 溢出,可以通过以下方法:

(1) 关闭计数器;

(2) 周期性的复位计数器:向看门狗关键字寄存器 WDKEY 先写入 0x55,然后再写入 0xAA,完成一次复位。

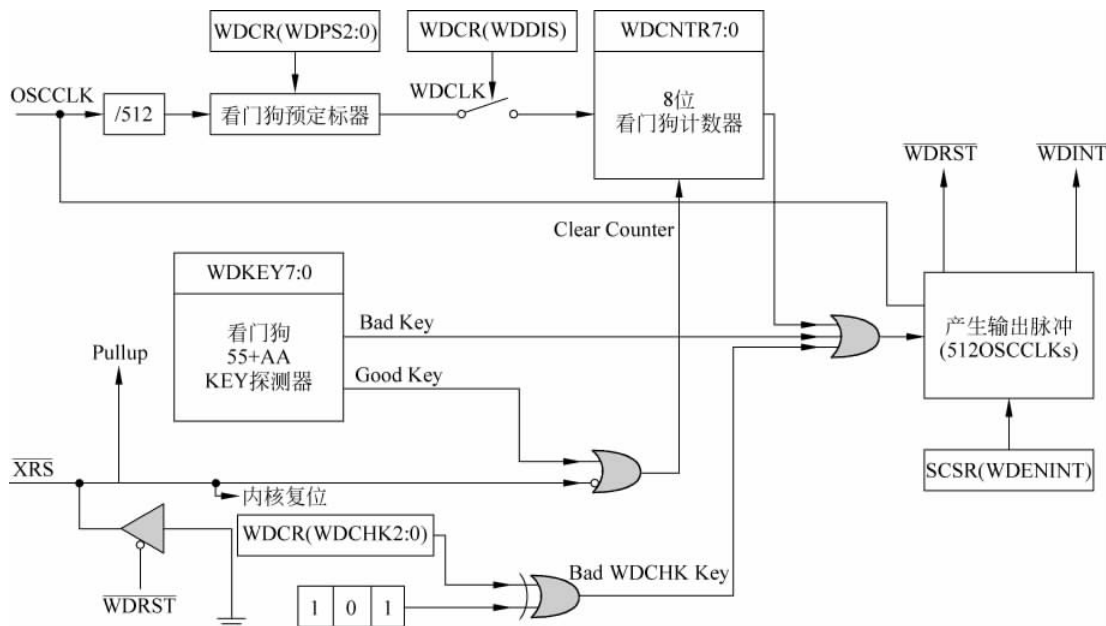


图 5-4 看门狗电路的结构框图

注意:写入的先后顺序不能变换。

2. WDCNTR 溢出操作

如果计数器溢出,要通过配置 SCSR 寄存器复位看门狗或者中断看门狗:

(1) 复位模式: 如果看门狗电路用来复位器件,当看门狗计数器溢出时,WDRST 信号将拉低 512 个 OSCCLK 周期。

(2) 中断模式: 如果看门狗用来产生中断,当看门狗计数器溢出时,WDINT 将被拉低 512 个 OSCCLK 周期,从而 WAKEINT 被触发。因为看门狗中断是在 WDINT 信号的下降沿触发,所以在 WDINT 信号返回高电平之前,WAKEINT 中断被重新使能,将不能立即获得另一个中断。

如果看门狗从中断模式到复位模式重新复位,并且 WDINT 仍为低电平,则器件立即被复位。可以通过读取 SCSR 寄存器的 WDINTS 位来判断 WDINT 信号当前状态。

3. 看门狗的低功耗模式

F2833x 一共有 3 种低功耗模式,在这 3 种低功耗模式下看门狗的运行情况如下所述:

(1) 在 STANDBY 模式下,CPU 的输入时钟被关闭,这使所有来自 SYSCLKOUT 的时钟都被关闭,但看门狗使用的是 OSCCLK 时钟信号,所以它仍处于正常工作。WDINT 信号被送入低功耗控制模块,可将器件从 STANDBY 模式下唤醒。

(2) 在 IDLE 模式下,看门狗的中断信号(WDINT)可以向 CPU 产生一个中断,使其从 IDLE 模式中被唤醒。

(3) 在 HALT 模式下,OSC 和 PLL 都停止工作,看门狗的时钟信号也被关闭,所以,唤醒工作都不能使用。

5.5 时钟和系统控制模块寄存器

表 5-3 列举了锁相环 PLL、时钟、看门狗功能和低功耗模式。

表 5-3 锁相环 PLL、时钟、看门狗功能和低功耗模式配置寄存器

名 称	地 址	长度(×16 位)	功 能 描 述
PLLSTS	0x00007011	1	PLL 状态寄存器
Reserved	0x00007012 ~0x00007018	7	保留寄存器
HISPCP	0x0000701A	1	高速外设时钟信号(HSPCLK)寄存器
LOSPCP	0x0000701B	1	低速外设时钟信号(LSPCLK)寄存器
PCLKCR0	0x0000701C	1	外设时钟控制寄存器 0
PCLKCR1	0x0000701D	1	外设时钟控制寄存器 1
LPMCR0	0x0000701E	1	低功耗模式控制寄存器 0
Reserved	0x0000701F	1	保留寄存器
PCLKCR3	0x00007020	1	外设时钟控制寄存器 3
PLLCR	0x00007021	1	锁相环 PLL 控制寄存器
SCSR	0x00007022	1	系统控制与状态寄存器
WDCNTR	0x00007023	1	看门狗计数器寄存器
Reserved	0x00007024	1	保留寄存器
WDKEY	0x00007025	1	看门狗复位关键字寄存器

续表

名 称	地 址	长度(×16 位)	功 能 描 述
Reserved	0x00007026 ~0x00007028	3	保留寄存器
WDCR	0x00007029	1	看门狗控制寄存器
Reserved	0x0000702A ~0x0000702F	6	保留寄存器

注意：PLL 控制寄存器(PLLCR)和 PLL 状态寄存器(PLLSTS)可以利用 XRS 或看门狗复位信号进行复位。

PCLKCR0/1/3 寄存器使能/禁止各种外设模块的输入时钟信号。PCLKCR0/1/3 寄存器的写操作会产生 2 个 SYSCLKOUT 周期延迟,这个延迟在访问外设配置寄存器之前将被计数。由于外设/GPIO 是复用的,所有的外设不能同时使用;然而同时开启外设的时钟信号是可以的,只是这样的配置是无效的。图 5-5 和表 5-4 分别给出了外设时钟控制寄存器 0 的各位功能定义及该寄存器中各位的相关描述。

15	14	13	12	11	10	9	8
ECANB ENCLK	ECANA ENCLK	McBSPB ENCLK	McBSPA ENCLK	SCIB ENCLK	SCIA ENCLK	Reserved	SPIA ENCLK
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R-0	R/W-0
7	6	5	4	3	2	1	0
Reserved		SCIC ENCLK	I ² CA ENCLK	ADC ENCLK	TBCLK SYNC	Reserved	
R-0		R/W-0	R/W-0	R/W-0	R/W-0	R-0	

图 5-5 外设时钟控制寄存器 0(PCLKCR0)

表 5-4 外设时钟控制寄存器 0 的字段描述

位	字 段	功 能 描 述
15	ECANBENCLK	ECAN-B 时钟使能位: 0—模块未计时(默认值)/1—模块 ECAN-B 使能并利用 SYSCLKOUT/2 计时
14	ECANAENCLK	ECAN-A 时钟使能位: 0—模块未计时(默认值)/1—模块 ECAN-A 使能并利用 SYSCLKOUT/2 计时
13	McBSPBENCLK	McBSPB-B 时钟使能位: 0—模块未计时(默认值)/1—模块 McBSPB-B 使能并利用低速时钟 LSPCLK 计时
12	McBSPAENCLK	McBSPB-A 时钟使能位: 0—模块未计时(默认值)/1—模块 McBSPB-A 使能并利用低速时钟 LSPCLK 计时
11	SCIBENCLK	SCI-B 时钟使能位: 0—模块未计时(默认值)/1—模块 SCI-B 使能并利用低速时钟 LSPCLK 计时
10	SCIAENCLK	SCI-A 时钟使能位: 0—模块未计时(默认值)/1—模块 SCI-A 使能并利用低速时钟 LSPCLK 计时
9	Reserved	保留
8	SPIAENCLK	SPI-A 时钟使能位: 0—模块未计时(默认值)/1—模块 SPI-A 使能并利用低速时钟 LSPCLK 计时

续表

位	字 段	功 能 描 述
7~6	Reserved	保留
5	SCICENCLK	SCI-C 时钟使能位：0—模块未计时(默认值)/1—模块 SCI-C 使能并利用低速时钟 LSPCLK 计时
4	I ² CAENCLK	I ² C 时钟使能位：0—模块未计时(默认值)/1—模块 I ² C 使能并利用 SYSCCLKOUT 计时
3	ADCENCLK	ADC 时钟使能位：0—模块未计时(默认值)/1—模块 ADC 使能并利用高速时钟 HSPCLK 计时
2	TBCLKSYNC	ePWM 模块基准时钟同步,允许用户将所有使能的 ePWM 模块与基准时钟同步： 0—每一个被使用的 ePWM 模块的基准时钟停止(默认值)。但如果 ePWM 模块时钟使能位在 PCLKCR1 寄存器中,那么 ePWM 模块将一直通过 SYSCCLKOUT 计数,即使 TBCLKSYNC 的值为 0/ 1—所有被使能的 ePWM 模块的时钟都从 TBCLK 的一个上升沿开始。为了完全同步,ePWM 模块的 TBCLK 寄存器的预置位必须被设置成同步。 ePWM 模块使用时钟的程序如下： • 在 PCLKCR1 寄存器中使能 ePWM 模块； • 设置 TBCLKSYNC 的值为 0； • 设置预定寄存器的值和 ePWM 模块； • 设置 TBCLKSYNC 的值为 1
1~0	Reserved	保留

图 5-6 和表 5-5 分别给出了外设时钟控制寄存器 1 的各位功能定义及该寄存器中各位的相关描述。

15	14	13	12	11	10	9	8
EQEP2 ENCLK	EQEP1 ENCLK	ECAP6 ENCLK	ECAP5 ENCLK	ECAP4 ENCLK	ECAP3 ENCLK	ECAP2 ENCLK	ECAP1 ENCLK
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
7	6	5	4	3	2	1	0
Reserved		EPWM6 ENCLK	EPWM5 ENCLK	EPWM4 ENCLK	EPWM3 ENCLK	EPWM2 ENCLK	EPWM1 ENCLK
R-0		R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0

图 5-6 外设时钟控制寄存器 1(PCLKCR1)

表 5-5 外设时钟控制寄存器 1 的字段描述

位	字 段	功 能 描 述
15	EQEP2ENCLK	EQEP2 时钟使能位：0—模块未计时(默认值)/1—模块 EQEP2 使能并利用 SYSCCLKOUT 计时
14	EQEP1ENCLK	EQEP1 时钟使能位：0—模块未计时(默认值)/1—模块 EQEP1 使能并利用 SYSCCLKOUT 计时
13	ECAP6ENCLK	eCAP6 时钟使能位,没有该模块时,该位保留：0—模块未计时(默认值)/1—模块 eCAP6 使能并利用 SYSCCLKOUT 计时
12	ECAP5ENCLK	eCAP5 时钟使能位,没有该模块时,该位保留：0—模块未计时(默认值)/1—模块 eCAP5 使能并利用 SYSCCLKOUT 计时

续表

位	字 段	功 能 描 述
11	ECAP4ENCLK	eCAP4 时钟使能位：0—模块未计时(默认值)/1—模块 eCAP4 使能并利用 SYSCLKOUT 计时
10	ECAP3ENCLK	eCAP3 时钟使能位：0—模块未计时(默认值)/1—模块 eCAP3 使能并利用 SYSCLKOUT 计时
9	ECAP2ENCLK	eCAP2 时钟使能位：0—模块未计时(默认值)/1—模块 eCAP2 使能并利用 SYSCLKOUT 计时
8	ECAP1ENCLK	eCAP1 时钟使能位：0—模块未计时(默认值)/1—模块 eCAP1 使能并利用 SYSCLKOUT 计时
7~6	Reserved	保留
5	EPWM6ENCLK	ePWM6 时钟使能位：0—模块未计时(默认值)/1—模块 ePWM6 使能并利用 SYSCLKOUT 计时
4	EPWM5ENCLK	ePWM5 时钟使能位：0—模块未计时(默认值)/1—模块 ePWM5 使能并利用 SYSCLKOUT 计时
3	EPWM4ENCLK	ePWM4 时钟使能位：0—模块未计时(默认值)/1—模块 ePWM4 使能并利用 SYSCLKOUT 计时
2	EPWM3ENCLK	ePWM3 时钟使能位：0—模块未计时(默认值)/1—模块 ePWM3 使能并利用 SYSCLKOUT 计时
1	EPWM2ENCLK	ePWM2 时钟使能位：0—模块未计时(默认值)/1—模块 ePWM2 使能并利用 SYSCLKOUT 计时
0	EPWM1ENCLK	ePWM1 时钟使能位：0—模块未计时(默认值)/1—模块 ePWM1 使能并利用 SYSCLKOUT 计时

图 5-7 和表 5-6 分别给出了外设时钟控制寄存器 3 的各位功能定义及该寄存器中各位的相关描述。

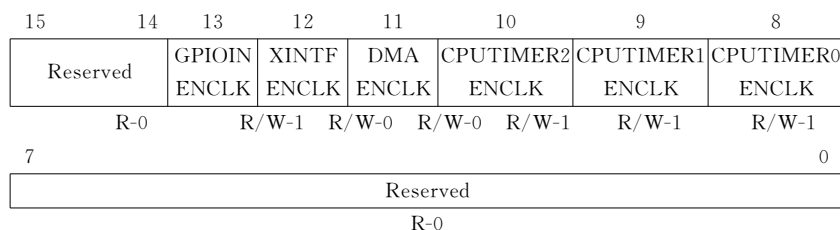


图 5-7 外设时钟控制寄存器 3(PCLKCR3)

表 5-6 外设时钟控制寄存器 3 的字段描述

位	字 段	功 能 描 述
15~14	Reserved	保留
13	GPIOINENCLK	GPIO 输入时钟使能位：0—模块未计时/1—模块 GPIO 使能并利用 SYSCLKOUT 计时(默认值)
12	XINTFENCLK	外部接口时钟使能位：0—模块未计时(默认值)/1—外部接口使能并利用 SYSCLKOUT 计时
11	DMAENCLK	DMA 时钟使能位：0—模块未计时(默认值)/1—DMA 使能并利用 SYSCLKOUT 计时

续表

位	字 段	功 能 描 述
10	CPUTIMER2ENCLK	CPUTIMER2 输入时钟使能位：0—模块未计时/1—模块 CPUTIMER2 使能并利用 SYSCLKOUT 计时(默认值)
9	CPUTIMER1ENCLK	CPUTIMER1 输入时钟使能位：0—模块未计时/1—模块 CPUTIMER1 使能并利用 SYSCLKOUT 计时(默认值)
8	CPUTIMER0ENCLK	CPUTIMER0 输入时钟使能位：0—模块未计时/1—模块 CPUTIMER0 使能并利用 SYSCLKOUT 计时(默认值)
7~0	Reserved	保留

图 5-8 和表 5-7 分别给出了高速外设时钟预定标的各位功能定义及该寄存器中各位的相关描述。

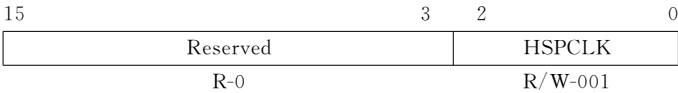


图 5-8 高速外设时钟预定标寄存器(HSPCP)

表 5-7 高速外设时钟预定标寄存器的字段描述

位	字 段	功 能 描 述
15~3	Reserved	保留
2~0	HSPCLK	用来配置高速外设时钟(HSPCLK)与系统时钟(SYSCLKOUT)之间的相互关系： 如果 HSPCP=0,那么 HSPCLK=SYSCLKOUT；如果 HSPCP≠0,那么 HSPCLK=SYSCLKOUT/(HSPCP×2)。 000：HSPCLK=SYSCLKOUT/1； 001：HSPCLK=SYSCLKOUT/2(复位后的默认值)； 010~111：HSPCLK=SYSCLKOUT/(HSPCP×2)

图 5-9 和表 5-8 分别给出了低速外设时钟预定标的各位功能定义及该寄存器中各位的相关描述。



图 5-9 低速外设时钟预定标寄存器(LOSPCP)

表 5-8 低速外设时钟预定标寄存器的字段描述

位	字 段	功 能 描 述
15~3	Reserved	保留
2~0	LSPCLK	用来配置高速外设时钟(LSPCLK)与系统时钟(SYSCLKOUT)之间的相互关系。 如果 LOSPCP=0,那么 LSPCLK=SYSCLKOUT；如果 LOSPCP≠0,那么 LSPCLK=SYSCLKOUT/(LOSPCP×2)。 000：LSPCLK=SYSCLKOUT/1； 001：LSPCLK=SYSCLKOUT/2(复位后的默认值)； 010~111：LSPCLK=SYSCLKOUT/(LOSPCP×2)

系统控制和状态寄存器(SCSR)包含看门狗受保护位和看门狗中断使能/禁止位。SCSR 的寄存器各位的功能描述如图 5-10 和表 5-9 所示。

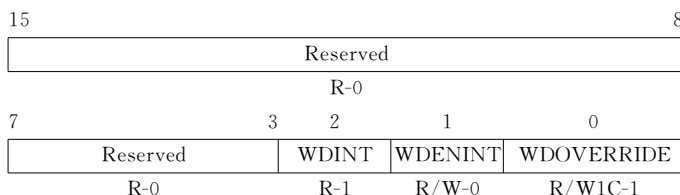


图 5-10 系统控制和状态寄存器(SCSR)

表 5-9 系统控制和状态寄存器(SCSR)位描述

位	字段	功 能 描 述
15~0	Reserved	保留
2	WDINTS	看门狗中断状态位。WDINTS 反映看门狗模块信号的当前状态,WDINTS 在状态之后的 2 个 SYSCLKOUT 周期。 如果看门狗中断用于将设备从 IDLE 或 STANDBY 模式中唤醒,在再次进入 IDLE 或 STANDBY 模式之前,可以借助此位来判断 $\overline{\text{WDINT}}$ 信号是否恢复为高电平。 0: 看门狗中断信号为有效电平; 1: 看门狗中断信号为无效电平
1	WDENINT	看门狗中断使能位。 0: 看门狗复位($\overline{\text{O}}$)输出信号被使能,看门狗中断($\overline{\text{O}}$)输出信号被禁止(默认复位状态)。当看门狗中断产生时, $\overline{\text{WDRST}}$ 信号在 512 个 OSCCLK 周期内持续为低电平,此时 $\overline{\text{WDINT}}$ 信号没有作用。 1: 看门狗复位($\overline{\text{O}}$)输出信号被禁止,看门狗中断($\overline{\text{O}}$)输出信号被使能(默认复位状态)。当看门狗中断产生时, $\overline{\text{WDINT}}$ 信号在 512 个 OSCCLK 周期内持续为低电平,此时 $\overline{\text{WDRST}}$ 信号没有作用
0	WDOVERRIDE	看门狗受保护位。 0: 写 0 没有影响。当该位被清除时,该位将保持此状态直到下一个复位发生。该位的当前状态可以被用户读取。 1: 可以改变看门狗控制寄存器(WDCR)中看门狗禁止位的状态。如果通过写 1 清除了该位,则不能修改 WDDIS 位

如图 5-11 和表 5-10 所示为看门狗计数寄存器位信息和功能介绍。

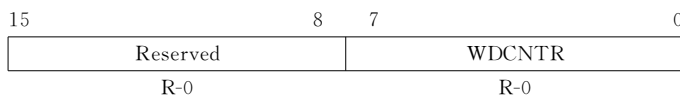


图 5-11 看门狗计数寄存器(WDCNTR)

表 5-10 看门狗计数寄存器(WDCNTR)位描述

位	字 段	功 能 描 述
15~8	Reserved	保留
7~0	WDCNTR	这些位为看门狗计数器的当前值。8 位计数器根据看门狗时钟(WDCLK)连续递增,如果计数器溢出,那么看门狗发出一个复位信号,如果向WDKEY 寄存器写一个有效组合,那么计数器将清零,看门狗的时钟基准由WDCR 寄存器配置

如图 5-12 和表 5-11 所示为看门狗复位寄存器位信息和功能介绍。

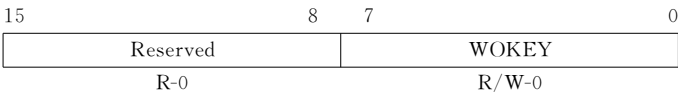


图 5-12 看门狗复位寄存器(WDKEY)

表 5-11 看门狗复位寄存器(WDKEY)位描述

位	字 段	值	功 能 描 述
15~8	Reserved		保留
7~0	WDKEY	0x55+0xAA 其他值	写入 0x55,紧接着写入 0xAA,使 WDCNTR 位清零; 写入 0x55 或 0xAA 之外的其他值将不会产生影响,但在 0x55 之后写入除 0xAA 外的其他值,那么需要重写 0x55; 读操作将返回 WDCR 寄存器的值

如图 5-13 和表 5-12 所示为看门狗控制寄存器位信息和功能介绍。

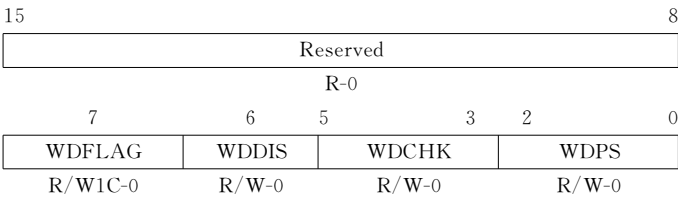


图 5-13 看门狗控制寄存器(WDCR)

表 5-12 看门狗控制寄存器(WDCR)位功能描述

位	字 段	功 能 描 述
15~8	Reserved	保留
7	WDFLAG	看门狗复位状态标志位。 0: 表示外部设备或上电复位条件。该位将一直锁存直到写 1 到 WDFLAG,将该位清零,写入 0 没有影响。 1: 表示看门狗复位(WDRST)满足复位条件
6	WDDIS	看门狗禁止位,复位时,看门狗模块激活。 0: 使能看门狗模块。只有当 SCSR 寄存器中的 WDOVERRIDE 位设为 1 时,WDDIS 的值才能被修改。 1: 禁止看门狗模块

续表

位	字 段	功 能 描 述
5~3	WDCHK	看门狗检查位。必须一直向这些位写 1、0、1,写入其他值设备会复位。 当看门狗被使能时,写入其他值会立即使设备复位或引起看门狗中断,读操作返回 0、0、0
2~0	WDPS	用于配置看门狗计数时钟的速率(相对于 OSCCLK/512); 000: WDCLK=OSCCLK/512/1; 其他: WDCLK=OSCCLK/512/2 ^{[2,0]-1}

5.6 时钟系统基本设置的编程例程

下面为一段看门狗基本设置例程,它反映了时钟系统设置的一种方式,通过该例程可以学习时钟系统编程设置的基本方式和各个设置要点。

```
#include "DSP2833x_Device.h"    //Headerfile Include File
#include "DSP2833x_Examples.h"  //Examples Include File
interrupt void wakeint_isr(void); //Prototype statements for functions found within this file
Uint32 WakeCount;
Uint32 LoopCount;
void main(void)
{
    InitSysCtrl();    //初始化系统函数
    DINT;             //Disable CPU interrupts; Clear all interrupts and initialize PIE vector table
    InitPieCtrl();    //初始化 PIE 控制寄存器,即: 禁止所有 PIE 中断使能,清零所有 PIE 中断标志位
    IER=0x0000;
    IFR=0x0000;      //这两句命令用来禁止所有 CPU 中断使能,清零所有 CPU 中断标志位
    InitPieVectTable(); //初始化 PIE 中断向量表
    EALLOW;          //This is needed to write to EALLOW protected registers
    PieVectTable.WAKEINT=&wakeint_isr; //使得 PieVectTable.WAKEINT 指向中断向量表
    EDIS;             //This is needed to disable write to EALLOW protected registers
    WakeCount=0;      //中断次数计数
    LoopCount=0;      //空闲循环计数

    EALLOW;
    SysCtrlRegs.SCSR=BIT1; //WDENINT 写 1,使能看门狗中断
    EDIS;

    PieCtrlRegs.PIECTRL.bit.ENPIE=1; //使能 PIE 模块
    PieCtrlRegs.PIEIER1.bit.INTx8=1; //使能 PIE 第一组中断向量
    IER |= M_INT1; //使能 CPU 中断 1
    EINT;          //Enable Global Interrupts
    ServiceDog();  //复位看门狗定时器
    EALLOW;
    SysCtrlRegs.WDCR=0x0028; //使能看门狗
    EDIS;

    for(;;)
    {
```

```
        LoopCount++;  
    }  
}  
  
interrupt void wakeint_isr(void)                //中断子程序  
{  
    WakeCount++;  
    PieCtrlRegs.PIEACK.all=PIEACK_GROUP1;    //使能模块的下次中断请求  
}
```

习题与思考

- 5-1 试画出时钟控制系统的系统结构框图。
- 5-2 试简述时钟及控制系统的主要功能。
- 5-3 试简述看门狗的功能及其工作原理。
- 5-4 试画出时钟系统框图(从外部时钟信号的生成到外设控制时钟的产生)。
- 5-5 试总结高低速时钟信号的设置相关知识要点。