



复旦微电子

FM33G0xx ***低功耗系列 MCU*** ***应用笔记***

LPTIMER 低功耗定时器

AN009

V1.0



本资料是为了让用户根据用途选择合适的上海复旦微电子集团股份有限公司（以下简称复旦微电子）的产品而提供的参考资料，不转让属于复旦微电子或者第三者所有的知识产权以及其他权利的许可。在使用本资料所记载的信息最终做出有关信息和产品是否适用的判断前，请您务必将所有信息作为一个整体系统来进行评价。

采购方对于选择与使用本文描述的复旦微电子的产品和服务全权负责，复旦微电子不承担采购方选择与使用本文描述的产品和服务的责任。除非以书面形式明确地认可，复旦微电子的产品不推荐、不授权、不担保用于包括军事、航空、航天、救生及生命维持系统在内的，由于失效或故障可能导致人身伤亡、严重的财产或环境损失的产品或系统中。未经复旦微电子的许可，不得翻印或者复制全部或部分本资料的内容。

今后日常的产品更新会在适当的时候发布，恕不另行通知。在购买本资料所记载的产品时，请预先向复旦微电子在当地的销售办事处确认最新信息，并请您通过各种方式关注复旦微电子公布的信息，包括复旦微电子的网站(<http://www.fmsh.com/>)。

如果您需要了解有关本资料所记载的信息或产品的详情，请与上海复旦微电子集团股份有限公司在当地的销售办事处联系。

商 标

上海复旦微电子集团股份有限公司的公司名称、徽标以及“复旦”徽标均为上海复旦微电子集团股份有限公司及其分公司在中国的商标或注册商标。

上海复旦微电子集团股份有限公司在中国发布，版权所有。

联系方式:

电表产品应用:

邢杰: xingjie@fmsh.com.cn TEL: 13916427310

陈钊: chenzhao@fmsh.com.cn TEL: 18616125501

水气热表及智能家居:

朱发旺: zhufawang@fmsh.com.cn TEL: 17749796664

姜涛: jiangtao@fmsh.com.cn TEL: 18701992908

超高频 900M 及物联网相关:

王晓腾: wangxiaoteng@fmsh.com.cn TEL: 13585663727

王天纵: wangtianzong@fmsh.com.cn TEL: 18221803903

资料下载及交流:

开发者论坛: <http://www.fmdevelopers.com.cn>



目 录

1 说明	1
2 原理及实现	1
2.1 说明	1
2.2 时钟源配置	2
2.2.1 LSCLK	2
2.2.2 RCLP	3
2.2.3 APB Bridge Clocks	3
2.2.4 LPTIN	3
2.3 工作模式	3
2.3.1 普通定时器	4
2.3.1 Timeout 模式	9
版本信息	14
上海复旦微电子集团股份有限公司销售及服务中心	15



图片目录

图 2-1 LPTIMER 相关时钟树	2
图 2-2 LSCLK 低频时钟	2
图 2-3 RCLP 低功耗内部环振.....	3
图 2-4 APB Bridge Clocks.....	3
图 2-5 LPTIM 原理图.....	5
图 2-6 LPTIM_PWM 示例	5
图 2-7 LPTIM_PWM 程序	7
图 2-8 PWM 波形输出	8
图 2-9 周期方波波形输出.....	9
图 2-10 TimeOut 模式示意图	10
图 2-11 LPTIM_TIMEOUT 示例.....	10
图 2-12 LPTIM_TIMEOUT 程序.....	12
图 2-13 LPTIM_TIMEOUT 仿真验证.....	13



1 说明

本文档为 FM33G0B 系列低功耗 MCU 的应用笔记，用于说明低功耗定时器 (LPTIM) 的应用方法及注意事项。FM33G0B 系列是复旦微电子公司开发的低功耗 MCU 芯片，请联系复旦微电子公司提供更多相关文档支持设计开发。

2 原理及实现

2.1 说明

LPTIM 是 16 位低功耗定时/计数器模块，该定时器内部通过异步电路实现低功耗性能。通过选择合适的工作时钟，LPTIM 在各种低功耗模式下保持运行，并且只消耗很低的功耗。LPTIM 甚至可以在没有内部时钟的条件下工作，因此可实现休眠模式下的外部脉冲计数功能。

LPTIM 的主要特性有：

- 16 位向上计数器
- 3 位异步时钟预分频器，8 种分频系数（1、2、4、8、16、32、64、128）
- 可选工作时钟
- 内部时钟源：LSCLK、RCLP、APB 时钟
- 外部时钟源：LPTIN（带有模拟滤波）
- 16 位比较寄存器
- 16 位目标值寄存器
- 软件/硬件触发
- 输入极性选择
- 无时钟外部脉冲计数
- 外部触发的休眠超时唤醒
- 支持 16 位 PWM

2.2 时钟源配置

LPTIM 时钟源有 4 个，分别为 LSCLK、RCLP、APB Bridge Clocks 及外部时钟源，其相关时钟系统如图 2-1 所示。

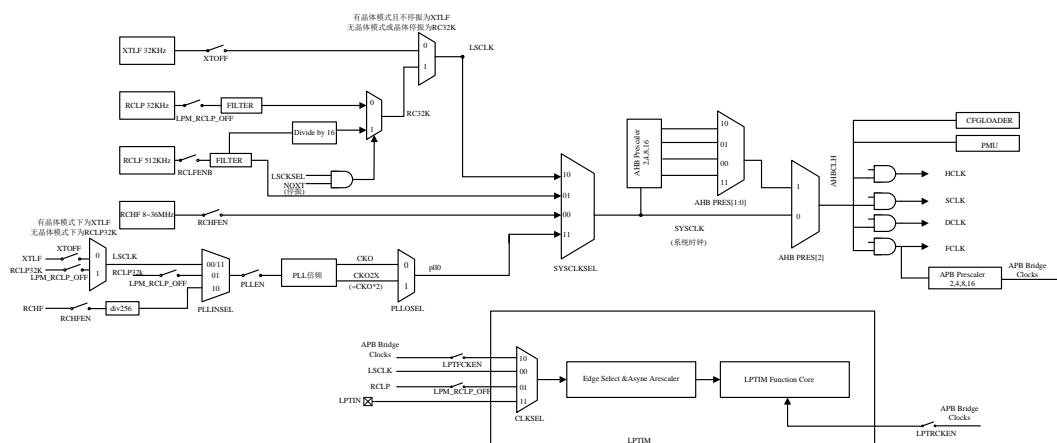


图 2-1 LPTIMER 相关时钟树

由时钟树我们可以得到 LPTIM 的 4 个时钟源的由来，如 2.2.1~2.2.4 所示。

2.2.1 LSCLK

LSCLK 为低频时钟，其时钟源可选择为外部 32k 晶体或内部 RC 振荡器,如图 2-2 所示。

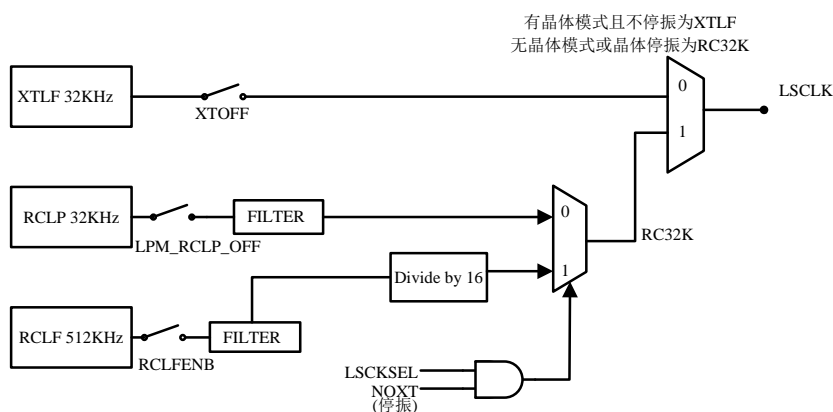


图 2-2 LSCLK 低频时钟

2.2.2 RCLP

RCLP 是低功耗内部环振,如如 2-3 所示。



图 2-3 RCLP 低功耗内部环振

2.2.3 APB Bridge Clocks

APB Bridge Clocks 由 SYSCLK 经分频得到，如图 2-4 所示。

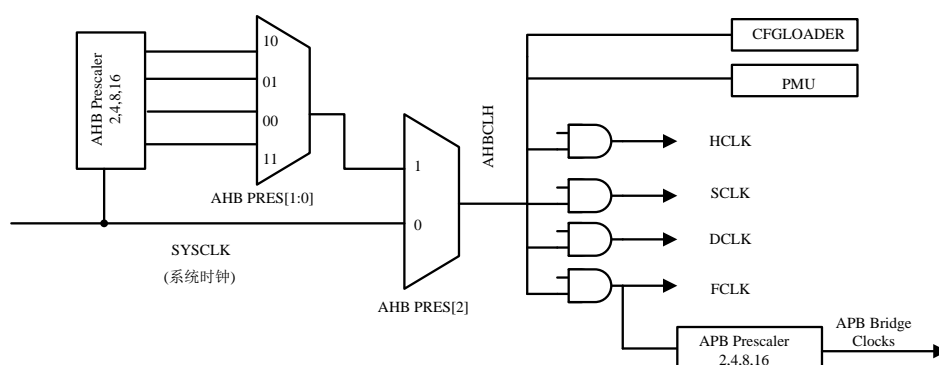


图 2-4 APB Bridge Clocks

2.2.4 LPTIN

LPTIN 是外部时钟源，内置 RC 滤波器，建议输入的时钟源频率不超过 10MHZ（RTCBKP 模式时，由于 GPIO 外设电源被关闭，所以该外部时钟源禁用）。

2.3 工作模式

LPTIM 有 4 种工作模式，分别为带波形输出的普通定时器模式、trigger 脉冲触发计数模式、外部异步脉冲计数模式、Timeout 模式。

2.3.1 普通定时器

当 LPTIM 作为普通定时器使用时，LPTIM 可以使用 LSCLK、RCLP 及 APB Bridge Clocks 作为输入时钟，不需要 External trigger。在作为普通定时器工作前，需要对定时器进行设置，必须设置的参数有：

- 时钟源选择（LSCLK、RCLP、APB Bridge Clocks）
- 3 位异步时钟预分频器，8 种分频系数（1、2、4、8、16、32、64、128）
- 定时器工作方式（带波形输出的普通定时器方式）
- 定时器计数模式（单次计数、连续计数）
- 脉宽调制方式（周期方波输出、PWM 输出）
- 极性选择（正极性波形，即第一次计数值=比较值时产生上升沿；负极性波形，即第一次计数值=比较值时产生下降沿）
- 比较值
- 目标值

带波形输出的普通定时器工作原理描述如下：

如下图 2-5 所示，在进行时钟选择及分频选择后，定时器获得计数时钟 LPTCLK，该时钟频率为 f ，周期为 T ，设置 16 位计数目标值 $target_value$ ，当定时器每经过定时周期 T 时计数值加 1，当计数到目标值 $target_value$ （等于 $target_value$ ），计数值清 0。

（1）当脉宽调制方式选择周期方波输出模式时，定时器计数到 $target_value$ 时 LPOUT 产生电平翻转，此时 LPOUT 产生的波形为占空比为 50% 的周期方波，频率为：

$$f_{OUT} = \frac{f}{2 \times (target_value + 1)}$$

此处 $target_value + 1$ 是基于防止出现 $\frac{f}{0}$ 的情况而设计的。

（2）当脉宽调制方式选择 PWM 输出模式时，除了目标值 $target_value$ ，还需要设置 16 位比较寄存器值 $compare_value$ ，该比较值决定了 PWM 模式下方波的占空比，当选择正极性波形输出，定时器计数值小于 $compare_value$ 时 LPOUT 输出低电平，大于 $compare_value$ 时 LPOUT 输出高电平，当计数到目标值 $target_value$ 时，计数值清 0，重新输出低电平。此时 LPOUT 产生的方波为占空比为：

$$duty_cycle = \frac{target_value - compare_value}{target_value + 1}$$

频率为：

$$f_{OUT} = \frac{f}{target_value + 1}$$

选择负极性输出则波形反向。

应用案例：

该模式的一个典型应用程序 LPTIM_PWM 如图 2-6 所示。

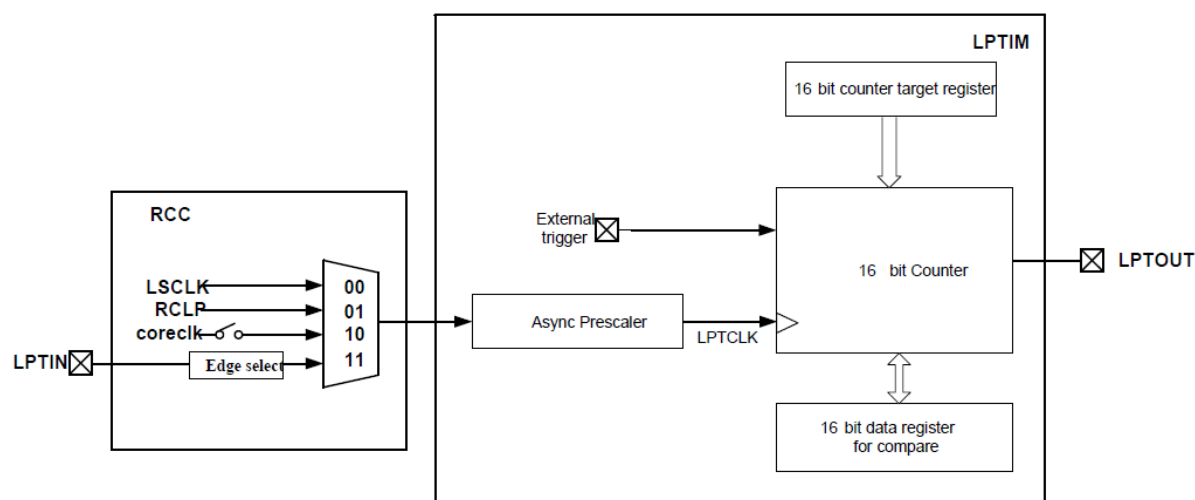


图 2-5 LPTIM 原理图

名称	修改日期	类型	大小
LPTIM PWM示例	2018/9/21 15:05	文件夹	
LPTIM TIMEOUT示例	2018/9/21 15:05	文件夹	

图 2-6 LPTIM_PWM 示例

打开该示例工程后，可以看到如图 2-7 所示程序。

主函数中首先调用函数 Init_System()对系统时钟进行了初始化，然后对 LPTIM 进行初始化，初始化包括 LPTIM 功能时钟及总线时钟使能、波形输出管脚 LPOUT 的配置及 LPTIM 参数设置。其中形输出管脚 LPOUT 为芯片的 PA12 引脚，该引脚波形信号可以用示波器验证。LPTIM 必须设置的参数本小节开头已经说明，由图 2-7 可以得到：

- 时钟源选择：LPTIM_LPTCFG_CLKSEL_RCLP，表示选择 RCLP 作为时钟源
- 3 位异步时钟预分频器：LPTIM_LPTCFG_DIVSEL_8，表示 RCLP 进行 8 分频，此时 LPTIM 时钟频率为



$$f = \frac{f_{RCLP}}{8} = \frac{32768}{8} = 4096Hz$$

- 定时器工作方式: LPTIM_LPTCFG_TMODE_PWMIM, 表示输出模式为带波形输出的普通定时器方式
- 定时器计数模式: LPTIM_LPTCFG_MODE_CONTINUE, 表示计数模式为连续计数, 单次计数和连续计数的区别在于单次计数模式在计数到目标值后计数值自动归 0 并自动停止, 连续模式自动归 0 后继续计数
- 脉宽调制方式: LPTIM_LPTCFG_PWM_PWM, 表示输出模式为 PWM 模式
- 极性选择: LPTIM_LPTCFG_POLARITY_POS, 表示计数值小于比较值时 LPOUT 输出低电平, 大于比较值时输出高电平, 此参数只在输出模式为 PWM 时必须设置
- 比较值 init_para.LPTIM_compare_value 和目标值 init_para.LPTIM_target_value, 其中比较值只在 PWM 模式下才必须需要设置。本示例中, 比较值为 200, 目标值为 1000, 根据前面的计算公式可以计算 LPOUT 输出频率:

$$f_{OUT} = \frac{f}{t_{\text{arget_value}} + 1} = \frac{4096}{1000 + 1} = 4.092Hz$$

周期为:

$$T_{OUT} = \frac{1}{f_{OUT}} = \frac{1}{4.092} = 244.4ms$$

占空比为:

$$duty_cycle = \frac{t_{\text{arget_value}} - compare_value}{t_{\text{arget_value}} + 1} = \frac{1000 - 200}{1000 + 1} = 79.9\%$$

在完成初始化后开启 LPTIM 模块, 即可产生 PWM, 产生的波形如图 2-8 所示。

```

1 #include "define_all.h"
2 uint32_t cnt_1k_Hz = 0;
3 void LPTIMER_COMM_TIMER_DEMO_init(void)//LPTIMER普通定时器示例程序初始化
4 {
5     LPTIM_InitTypeDef init_para;
6     //使能LPTIMER的外设时钟
7     RCC_PERCLK_SetableEx( LPTICLK, ENABLE ); //LPTIM功能时钟使能
8     RCC_PERCLK_SetableEx( LPTICLK, ENABLE ); //LPTIM总线时钟使能
9     RCC_PERCLK_SetableEx( PDCCLK, ENABLE ); //IO控制时钟寄存器使能
10    AltFunIO(GPIOA, GPIO_Pin_12, 0);//PA12, 数字特殊功能口, 普通
11    init_para.LPTIM_TMODE = LPTIM_LPTCFG_TMODE_PWMIM;//设置工作模式为带波形输出的普通定时器模式
12    init_para.LPTIM_MODE = LPTIM_LPTCFG_MODE_CONTINUE;//设置计数模式为连续计数模式
13    init_para.LPTIM_PWM = LPTIM_LPTCFG_PWM_PWM;//选择脉宽调制方式
14    init_para.LPTIM_POLAR = LPTIM_LPTCFG_POLARITY_POS;//第一次计数值=比较值是产生上升沿
15    init_para.LPTIM_TRIG_CFG = LPTIM_LPTCFG_TRIGCFG_POS;//外部输入信号上升沿trigger
16    init_para.LPTIM_FLTEN = ENABLE;//使能数字滤波
17    init_para.LPTIM_LPTIN_EDGE = LPTIM_LPTCFG_EDGESEL_POS;
18    init_para.LPTIM_CLK_SEL = LPTIM_LPTCFG_CLKSEL_RCLP;//选择LPTIMER的时钟源
19    init_para.LPTIM_CLK_DIV = LPTIM_LPTCFG_DIVSEL_8;//设置分频值
20    init_para.LPTIM_compare_value = 200;//设置比较值
21    init_para.LPTIM_target_value = 1000;//设置目标值
22    LPTIM_Init(&init_para);//初始化LPTIMER
23    //使能LPTIMER的外设中断
24    NVIC_DisableIRQ(LPTIM_IRQn);
25    NVIC_SetPriority(LPTIM_IRQn, 2);
26    NVIC_EnableIRQ(LPTIM_IRQn);
27 }
28
29 void LPTIMER_COMM_TIMER_DEMO_start(void)//DEMO启动
30 {
31     LPTIM_LPTIE_OVIE_Setable(ENABLE);//使能LPTIMER的溢出中断
32     LPTIM_LPTCTRL_LPTEN_Setable(ENABLE);//LPTIMER模块使能
33 }
34
35 void LPTIMER_COMM_TIMER_stop(void)//DEMO关闭
36 {
37     LPTIM_LPTIE_OVIE_Setable(DISABLE);//关闭LPTIMER的溢出中断
38     LPTIM_LPTCTRL_LPTEN_Setable(DISABLE);//LPTIMER模块关闭
39 }
40
41 void LPTIM_IRQHandler(void)//普通定时器中断服务程序
42 {
43     if (LPTIM_LPTIE_OVIE_Getable() == ENABLE)//溢出中断使能
44     {
45         if (LPTIM_LPTIF_OVIF_Chk() == SET)//溢出中断标志被置起
46         {
47             LPTIM_LPTIF_OVIF_Clr();//清除中断标志
48             cnt_1k_Hz ++;
49             if (cnt_1k_Hz >= 1000)
50             {
51                 cnt_1k_Hz = 0;
52                 LED0_TOG;
53             }
54         }
55     }
56 }
57
58
59
60
61
62
63 int main (void)
64 {
65     Init_System(); //系统初始化
66
67     LPTIMER_COMM_TIMER_DEMO_init();
68     LPTIMER_COMM_TIMER_DEMO_start();//启动
69
70     for( ; ; )
71     {
72         IWDI_Clr();
73         TicksDelayMs( 50, NULL );
74     }
75 }
76

```

图 2-7 LPTIM_PWM 程序

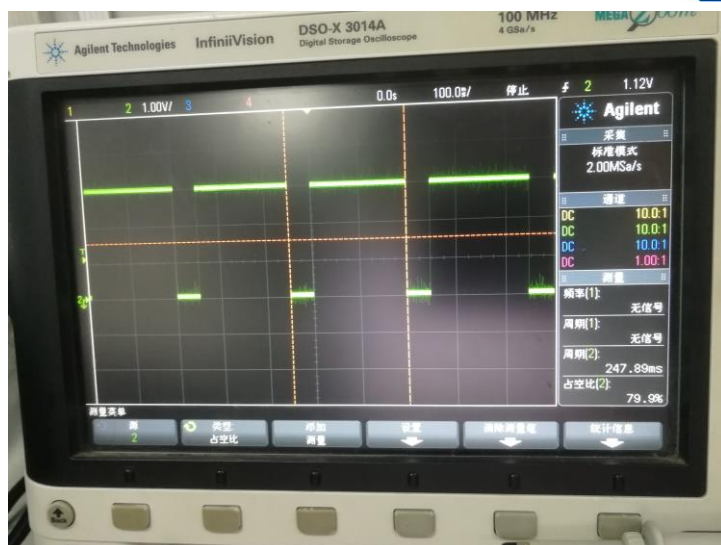


图 2-8 PWM 波形输出

由图可得，输出波形周期 247.89ms，占空比 79.9%，与计算结果基本一致。在该示例基础上，如果要以周期方波方式输出，则只需要将脉宽调制方式值选择为 LPTIM_LPTCFG_PWM_TOGGLE，该方式产生波形与 PWM 方式不同，前面已经介绍，产生的方波频率：

$$f_{OUT} = \frac{f}{2 * (target_value + 1)} = \frac{4096}{2 * 1001} = 2.046Hz$$

周期为：

$$T_{OUT} = \frac{1}{f_{OUT}} = \frac{1}{2.046} = 488.8ms$$

占空比恒定为 50%，波形如图 2-9 所示。



图 2-9 周期方波波形输出

2.3.1 Timeout 模式

当 LPTIM 模式为 Timeout 时，LPTIM 可以使用 LSCLK、RCLP、APB Bridge Clocks 及 LPTIN 作为输入时钟，需要 External trigger 输入。在 LPTIM 模式为 Timeout 时，需要对定时器进行设置，必须设置的参数有：

- 时钟源选择（LSCLK、RCLP、APB Bridge Clocks、LPTIN）
- 3 位异步时钟预分频器，8 种分频系数（1、2、4、8、16、32、64、128）
- 外部触发边沿选择（上升沿触发、下降沿触发、上升沿及下降沿触发）
- 定时器计数模式（单次计数、连续计数）
- 定时器工作方式（Timeout 模式）
- LPTIN 输入边沿选择（上升沿计数、下降沿计数）
- 目标值

需要注意的是，LPTIM 使能后，计数器并不会立即工作，必须由外部触发后计数器清零并启动计数。该模式原理如图 2-10 所示。

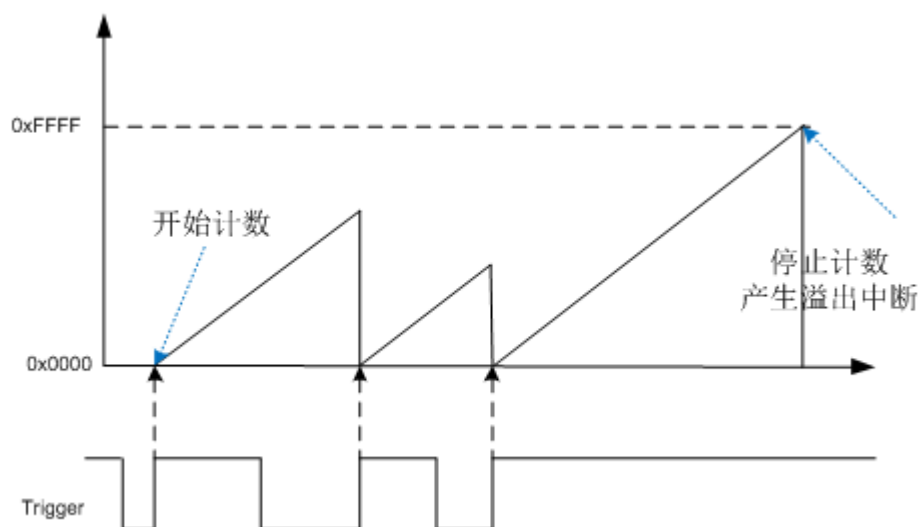


图 2-10 TimeOut 模式示意图

如图，每当 trigger 产生时，定时器从 0 开始计数，直到下一个 trigger 产生计数清零并重新计数。当计数到目标值仍未产生 trigger 时，计数器溢出，无论计数模式是否选择连续模式，LPTIM 都会被停止。

应用案例：

该模式的一个典型应用程序 LPTIM_TIMEOUT 如图 2-11 所示。

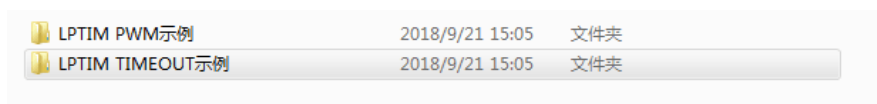


图 2-11 LPTIM_TIMEOUT 示例

打开该示例工程后，可以看到如图 2-12 所示程序。

主函数中首先调用函数 Init_System()对系统时钟进行了初始化，然后对 LPTIM 进行初始化，初始化包括 LPTIM 功能时钟及总线时钟使能、trigger 输入管脚 PA13 的配置及 LPTIM 参数设置。其中 trigger 由示波器产生方波信号作为触发信号。LPTIM 必须设置的参数本小节开头已经说明，由图 2-12 可以得到：

- 时钟源选择：LPTIM_LPTCFG_CLKSEL_PCLK，表示选择 APB Bridge Clocks 作为时钟源。
- 3 位异步时钟预分频器：LPTIM_LPTCFG_DIVSEL_32，表示 APB Bridge Clocks 进行 32 分频，此时 LPTIM 时钟频率为：

$$f = \frac{f_{APB}}{32} = \frac{8000000}{32} = 250000Hz$$

该时钟为 trigger 输入的采样时钟，由采样定理，trigger 输入时钟需满足：

$$f_{IN} \leq \frac{1}{2}f = 125000Hz$$

- 外部触发边沿选择：LPTIM_LPTCFG_TRIGCFG_POS，表示外部信号上升沿触发
- 定时器计数模式：LPTIM_LPTCFG_MODE_CONTINUE，表示计数模式为连续计数
- 定时器工作方式：LPTIM_LPTCFG_TMODE_TIMEOUT，表示工作模式为 Timeout，该模式下该值固定不变
- LPTIN 输入边沿选择：当用 LPTIN 作为输入时钟源时，还需要设置 LPTIN 的输入边沿，本示例选用 APB Bridge Clocks 作为输入时钟，因此该设置不是必须的
- 目标值：本示例设定目标值为 999，即溢出频率：

$$f_{overflow} = \frac{f}{t_{arg et_value} + 1} = \frac{250000}{1000} = 250Hz$$

表示一个溢出周期内必须 Trigger 一次，否则会产生溢出中断，即要进入溢出中断，需要满足：

$$f_{IN} \leq f_{overflow} = 250Hz$$

综上，在本示例参数下：

(1) 当 $f_{IN} \leq f_{overflow} = 250Hz$ 时，trigger 信号能被采集到，且定时器会进入溢出中断。

(2) 当 $125000Hz = \frac{1}{2}f > f_{IN} > f_{overflow} = 250Hz$ 时，trigger 信号能被采集到，且定时器不会进入溢出中断。

(3) 当 $f_{IN} \geq \frac{1}{2}f = 125000Hz$ 时，trigger 信号不能保证被全部采集到。

在完成初始化后开启 LPTIM 模块，即可采集到 trigger 信号，仿真验证如图 2-13 所示。

```

1 #include "define_all.h"
2 void LPTIMER_TIMEOUT_DEMO_GPIO_init(void) //GPIO初始化
3 {
4     RCC_PERCLK_SetableEx( PDCLK, ENABLE ); //IO控制时钟寄存器使能
5     AltFuncIO(GPIOA, GPIO_Pin_13, 0) //PA13, LPTIM, 数字特殊功能口, 普通
6 }
7 void LPTIMER_TIMEOUT_DEMO_init(void)
8 {
9     LPTIM_InitTypeDef init_para;
10    LPTIMER_TIMEOUT_DEMO_GPIO_init();
11    //使能LPTIMER外设时钟
12    RCC_PERCLK_SetableEx( LPTIMCLK, ENABLE ); //LPTIM功能时钟使能
13    RCC_PERCLK_SetableEx( LPTIMCLK, ENABLE ); //LPTIM系统时钟使能
14    init_para.LPTIM_MODE = LPTIM_LPTCFG_MODE_TIMEOUT; //设置工作模式为TIMEOUT
15    init_para.LPTIM_MODE = LPTIM_LPTCFG_MODE_CONTINUE; //设置计数模式为连续计数模式
16    init_para.LPTIM_PWN = LPTIM_LPTCFG_PWN_PWN; //选择PWM输出模式
17    init_para.LPTIM_POLAR = LPTIM_LPTCFG_POLARITY_POS; //产生输出波形上升沿
18    init_para.LPTIM_TRIG_CFG = LPTIM_LPTCFG_TRIGCFG_POS; //外部输入信号上升沿trigger
19    init_para.LPTIM_FLTEN = ENABLE; //使能数字滤波
20    init_para.LPTIM_LPTIM_EDGE = LPTIM_LPTCFG_EDGESEL_POS; //选择外溢时钟
21    init_para.LPTIM_CLK_SEL = LPTIM_LPTCFG_CLKSEL_PCLK; //选择外溢时钟
22    init_para.LPTIM_CLK_DIV = LPTIM_LPTCFG_DIVSEL_32; //32分频, 如果PCLK为8M, 则分频时钟为250k
23    init_para.LPTIM_compare_value = 0; //设置比较值
24    init_para.LPTIM_target_value = 999; //设置目标值, 本例设为1000, 即在1000次分频时钟之内必须被trig一下, 否则产生溢出中断
25    LPTIM_Init(&init_para); //初始化LPTIMER
26    //使能LPTIMER3外溢中断
27    NVIC_DisableIRQ(LPTIM_IRQn);
28    NVIC_SetPriority(LPTIM_IRQn, 2);
29    NVIC_EnableIRQ(LPTIM_IRQn);
30 }
31 void LPTIMER_TIMEOUT_DEMO_start(void) //DEMO启动
32 {
33     LPTIM_LPTIE_OVIE_Setable(ENABLE); //使能LPTIMER的溢出中断
34     LPTIM_LPTCTRL_LPTEN_Setable(ENABLE); //LPTIMER模块使能
35 }
36 void LPTIMER_TIMEOUT_DEMO_stop(void) //DEMO关闭
37 {
38     LPTIM_LPTIE_OVIE_Setable(DISABLE); //关闭LPTIMER的溢出中断
39     LPTIM_LPTCTRL_LPTEN_Setable(DISABLE); //LPTIMER模块关闭
40 }
41 void LPTIM_IRQHandler(void) //定时器中断服务程序
42 {
43     if (LPTIM_LPTIE_OVIE_Getable() == ENABLE) //溢出中断使能
44     {
45         if (LPTIM_LPTIF_OVIF_CK() == SET) //溢出中断标志被置起
46         {
47             LPTIM_LPTIE_OVIE_Clr(); //清除中断标志
48             LED0_ON; //点亮LED表示发生溢出
49         }
50         else //其他情况, 另外做相应处理
51         {
52         }
53     }
54 }
55
56 int main (void)
57 {
58     Init_System(); //系统初始化
59     LED0_OFF; //关闭LED
60     LPTIMER_TIMEOUT_DEMO_init();
61     LPTIMER_TIMEOUT_DEMO_start(); //启动
62
63     for(;;)
64     {
65         IWDI_Clr();
66         TicksDelayMs( 50, NULL );
67     }
68 }

```

图 2-12 LPTIM_TIMEOUT 程序

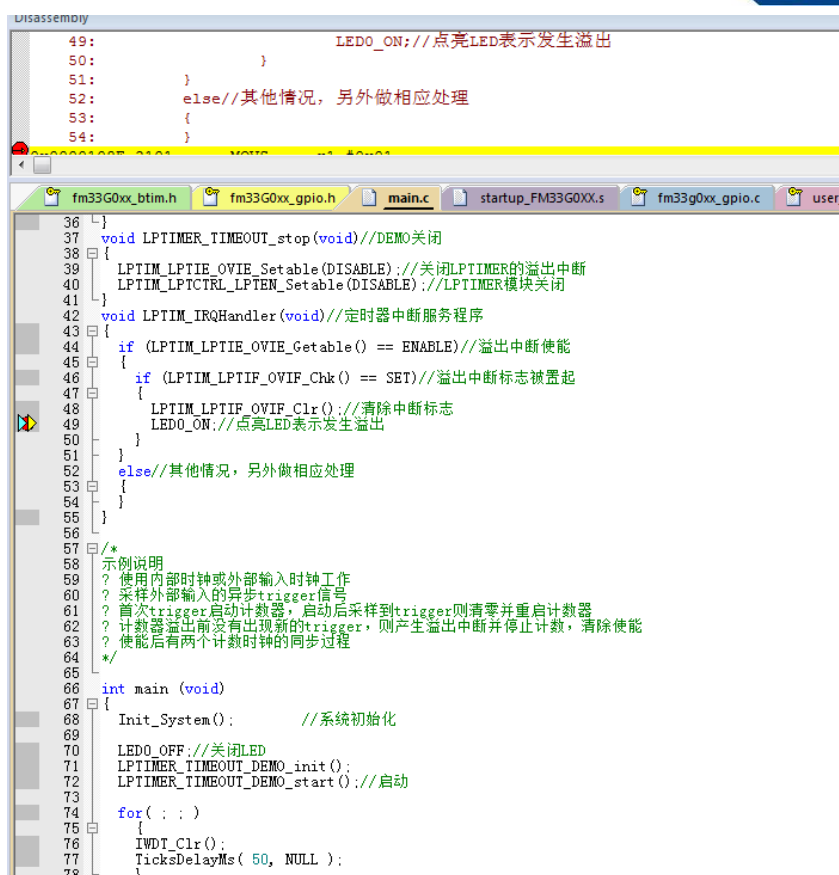


图 2-13 LPTIM_TIMEOUT 仿真验证

验证过程：由示例程序可知，LPTIM 的 trigger 信号输入管脚为 PA13，为使程序进入溢出中断，trigger 信号频率应不大于 250Hz，因此示波器选择输出 200Hz 方波信号到 PA13，当 PA13 采集到 trigger 信号时，定时器开始计数，并产生溢出，进入溢出中断，若未采集到 trigger 信号，定时器不工作，无法进入溢出中断，因此在溢出中断内打断点可以验证是否采集到 trigger 信号。如图 2-13 进入仿真设置断点后可以执行到断点处，即验证完成。



版本信息

版本号	发布日期	更改说明
1.0	2018.9	首次发布



上海复旦微电子集团股份有限公司销售及服务中心

上海复旦微电子集团股份有限公司

地址：上海市国泰路 127 号 4 号楼

邮编：200433

电话：(86-021) 6565 5050

传真：(86-021) 6565 9115

上海复旦微电子（香港）股份有限公司

地址：香港九龙尖沙咀东嘉连威老道 98 号东海商业中心 5 楼 506 室

电话：(852) 2116 3288 2116 3338

传真：(852) 2116 0882

北京办事处

地址：北京市东城区东直门北小街青龙胡同 1 号歌华大厦 B 座 423 室

邮编：100007

电话：(86-10) 8418 6608

传真：(86-10) 8418 6211

深圳办事处

地址：深圳市华强北路 4002 号圣廷苑酒店世纪楼 1301 室

邮编：518028

电话：(86-0755) 8335 0911 8335 1011 8335 2011 8335 0611

传真：(86-0755) 8335 9011

台湾办事处

地址：台北市 114 内湖区内湖路一段 252 号 12 楼 1225 室

电话：(886-2) 7721 1889

传真：(886-2) 7722 3888

新加坡办事处

地址：237, Alexandra Road, #07-01, The Alexcier, Singapore 159929

电话：(65) 6472 3688

传真：(65) 6472 3669

北美办事处

地址：2490 W. Ray Road Suite#2 Chandler, AZ 85224 USA

电话：(480) 857-6500 ext 18

公司网址：<http://www.fmsh.com/>