基于LGTSDK Builder

LGT8F690A 快速开发系列教程

第四篇：TIMER0的使用

© 2017 LogicGreen Technologies

TIMR0编程

本篇为系列教程的第四篇。如果需要了解教程相关的软件硬件环境，请参考本系列教程的第一篇：《LGT8F690A快速开发系列教程第一篇_急速上手》

LGT8F690的TIMR0是一个非常简单的8位定时器。TIMR0只有一个8位的计数器TMR0， 计数器TMR0在计数时钟的驱动下，累加计数。 当TMR0计数到溢出(从0xFF 到 0x00)时， 触发溢出中断事件。如果要改变TIMER0的溢出周期， 需要手动重载TMR0寄存器。

TIMER0的计数时钟有3种选择： 系统指令周期，外部T0CKI的上升沿或者下降沿。
TIMER0有一个和看门狗共享的系统预分频器。 同一时刻，该预分频器只能分配给其中一个。

由于TIMER0如此简单， 基本上是用来触发一个周期性的事件使用。
这里， 我们将使用TIMER0重新实现点灯伟业。 但这一次不同的是， 我们要点一个呼吸灯！

TIMER0的呼吸灯

呼吸灯的原理就是实现一个渐明渐暗的节奏变化， 明暗的变化可以通过控制输出频率的占空比实现。可以说，闪灯是固定的占空比；呼吸灯是用渐变的占空比！

TIMR0不能直接产生PWM，我们只用TIMER0产生一个固定的时基； PWM和占空比的变化将由软件实现。

下面简单说下关于定时器的定时时基确定方法：假定呼吸灯是1秒钟一次，PWM分辨率为256. 也就是说，1秒钟之内， 要跑完256个周期。 PWM的周期就是4ms左右。同时这4ms的周期分辨率是256, 也就是说一个最小分辨率是15.6us (4000us/256)。因此， 我们就将TIMER0的时基定位15.6us左右。

由于系统响应中断， 以及在中断服务中的处理， 会增加定时周期的长度， 因此，我们将会尽量将系统时钟跑的快一些。这里我们定义使用8MHz的主频。 由于TIMER0的计数时钟是指令周期，我们将指令周期设置为1T， 这样TIMR0的时钟源就是8MHz。

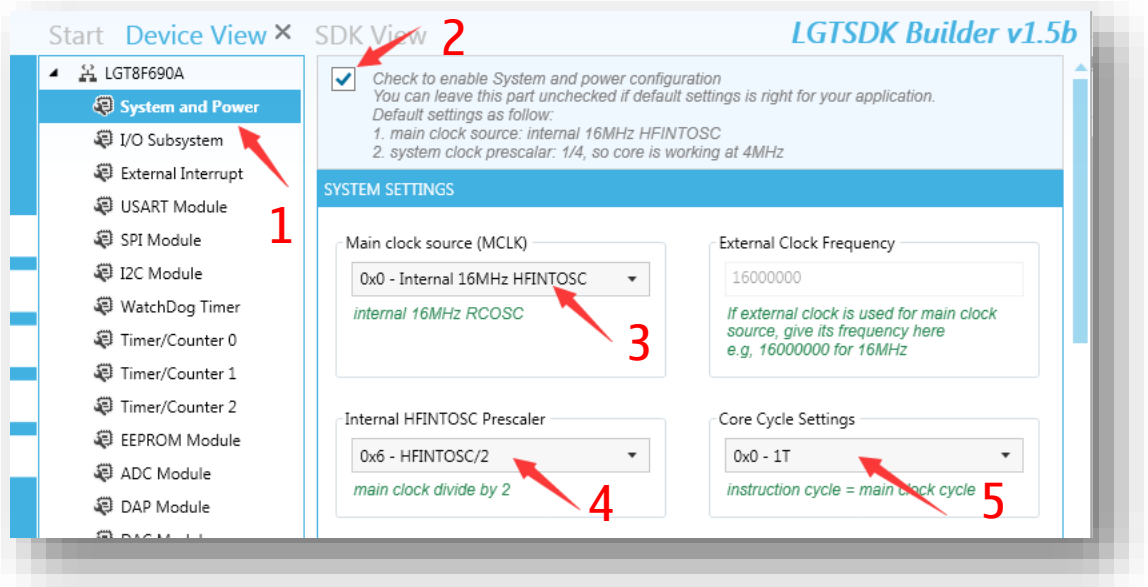
8MHz的周期是125ns， 产生15.6us的定时周期需要计数周期数为：15600/125 = 124
因此，计数器重载数值应该为：256 – 124 = 132 (注意TIMER0是溢出中断的)

我们还是使用RA0作为点灯的驱动信号。 LED等的接法和教程第一篇中的完全一致！
下面启动LGTSDK Builder， 开始配置我们所需的资源。

© 2017 LogicGreen Technologies

首先新建一个工程， 选择芯片， 封装。 工程命名：lgt8f690a_tc0

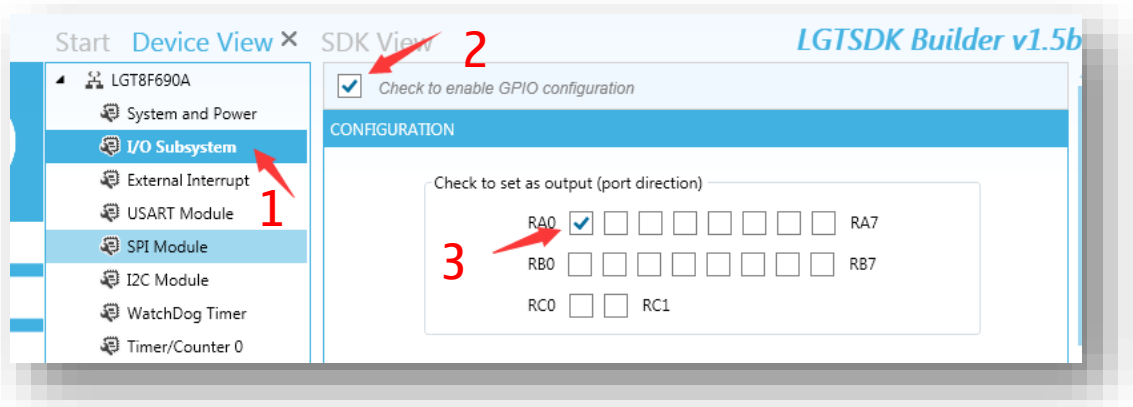
先来配置系统时钟，实现8MHz/1T



配置3： 选择内部16MHz RC 振荡器
配置4： 预分频选择2分频， 产生一个8MHz的系统主时钟
配置5： 内核周期为1T

注意： 此处4/5的配置， 优先级高于配置位的设置。

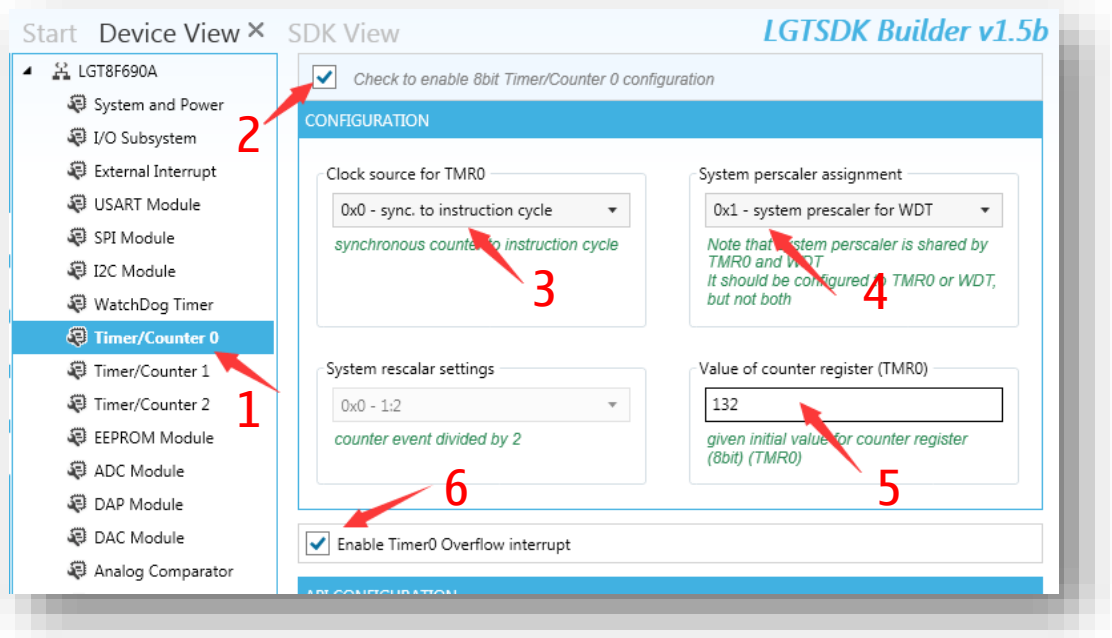
然后是I/O配置， 我们需要将RA0设置为输出I/O，用于驱动LED闪烁！



以上配置都在本系列的前几篇教程中介绍。 如有疑问， 请参考系列教程的二三篇。

接下来是配置本篇的主角， TIMER0：

© 2017 LogicGreen Technologies



配置3： 选择定时器0的计数时钟， 此处选择为指令周期
配置4： 系统预分频。此处不用，将预分频配置给WDT模块
配置5： 计数器的计数初始值，这个也就是我们之前计算的计数器重载的值
配置6： 使能定时器0的溢出中断功能

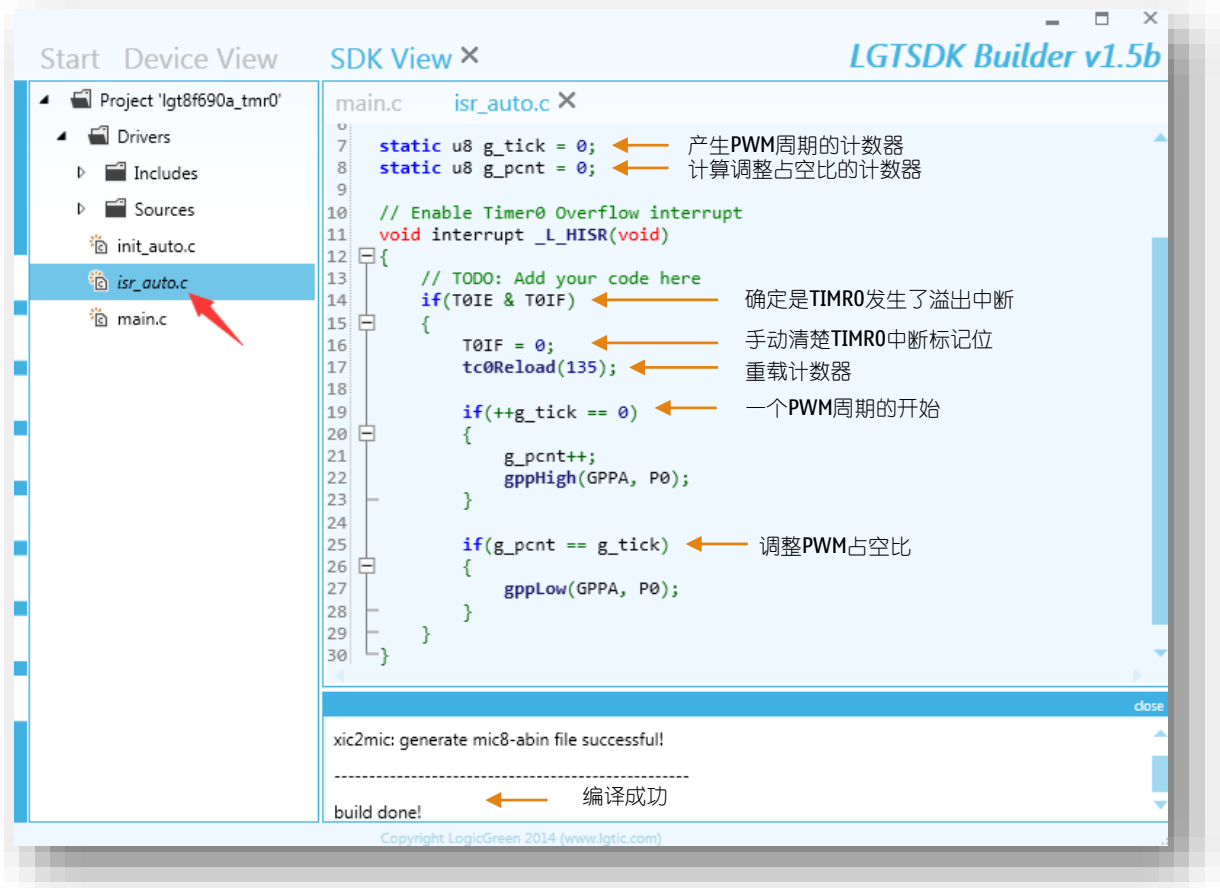
设置完成， 在 [Device View]界面下，点击[Build], 产生SDK源码！

因为我们使能了中断， 这是中断复位所在的文件。本篇教程的编码工作也都将在这个文件里面实现

© 2017 LogicGreen Technologies

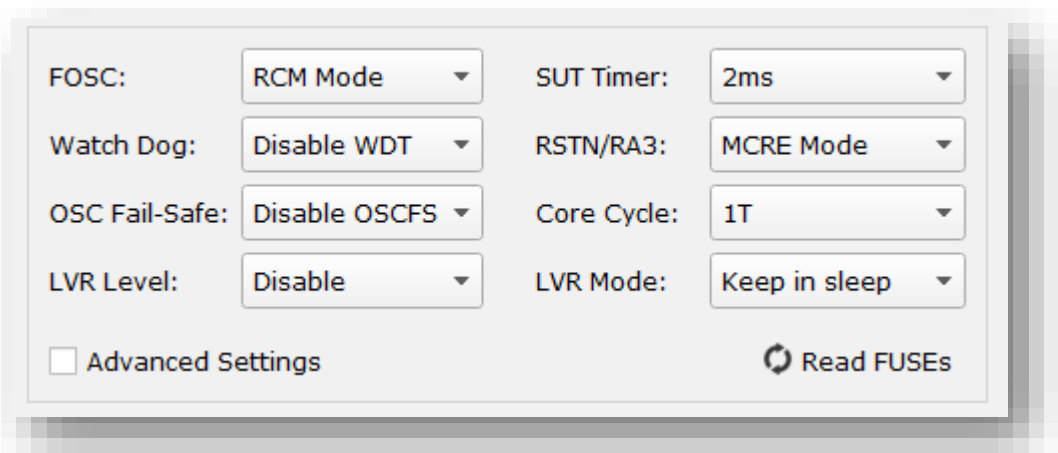
下面是呼吸灯的代码实现：
在工程管理窗口中双击isr_auto.c， 打开中断服务源文件。 可以看到里面有一个空的中断服务程序。我们需要把呼吸灯的代码写到这个中断服务中即可。

代码原理非常简单。 我们需要两个变量辅助产生PWM周期和调整占空比即可。
完整的代码如下：



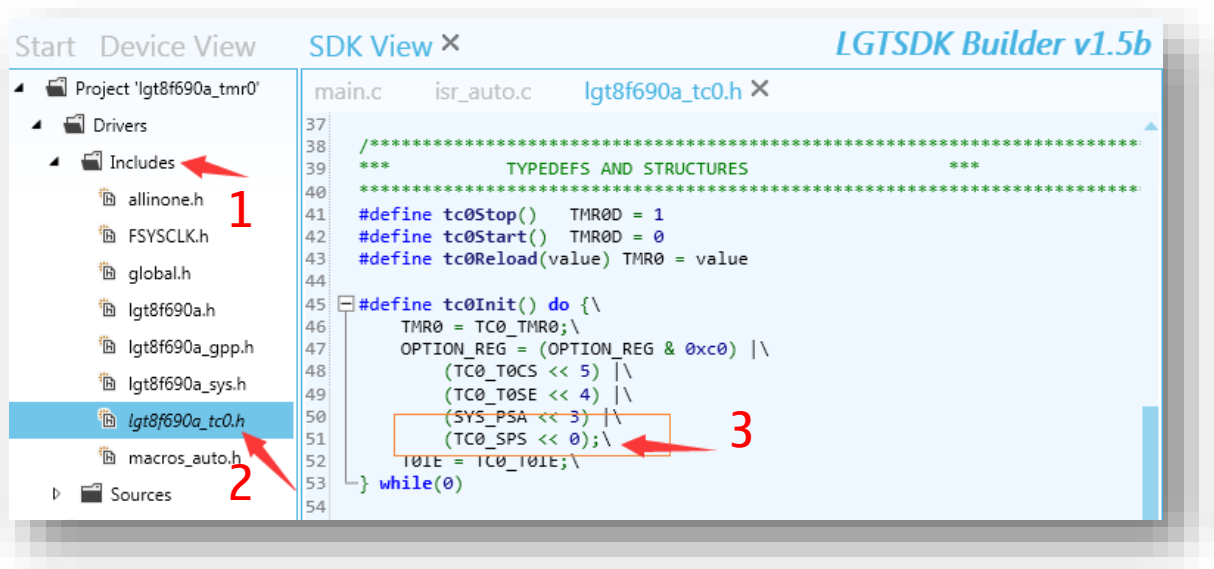
编译成功！ 可以下载到最小开发板上测试了！

使用LGTMix_ISP下载时， 需要确认下主时钟选择为内部RC就可以了。其他都默认即可。



© 2017 LogicGreen Technologies

LGTSDK Builder 1.5beta22版本， 在产生TIMER0的源代码时， 有如下BUG， 请在产生的源文件中做如下修改：



请将3处的代码， 将原先的 TC0_PSA 改为 TC0_SPS 即可！

此BUG也将会在后续版本中更新！

© 2017 LogicGreen Technologies