RTC\_SWTimer Application Notes

1. 本功能是通过RTC计数器实现精度为1ms的软件定时器，使用方法和功能效果类似于FreeRtos提供的sw timer功能。
2. 添加方法：在keil工程中添加newtimers.c编译即可，newtimers.h和list\_newtimer.h要放在include path里。
3. API说明：

* BaseType\_t newTimerCreateTimerTask(void); // timer task create，RTC\_INIT()放在这个函数里初始化，如果工程中也用到RTC，需要考虑二者的init时机。
* TimerHandle\_t newxTimerCreate(……..); // timer create,函数参数同FreeRtos的TimerHandle\_t xTimerCreate(……..)一致; timer设计为从heap中动态分配内存，一个timer占40byte，实际使用中需要考虑timer个数和内存的关系。
* newxTimerStart(TimerHandle\_t xTimer ); // start timer
* newxTimerStop(TimerHandle\_t xTimer ); // stop timer
* newxTimerDelete(TimerHandle\_t xTimer ); // delete timer，释放内存
* newxTimerChangePeriod(TimerHandle\_t xTimer, TickType\_t xNewPeriod );

1. Active mode下，精度可以达到10us以内的精度，进出Dlps会有3-4ms左右的误差。

改动：

问题1：在一个快到期的timer之前去start timer，rtc设置错误，导致8分钟后才响应：

解决：1. 在newInsertTimerInActiveList里去判断是否是第一个timer，如果不是，则不去set comp。增加一个变量（need check）

2. 在newProcessExpiredTimer对auto reload timer

的调用直接传false，即要进行rtc set\_comp。

3. 针对stop和delete timer也进行是否是first timer的判断。

4. 在set comp前后添加临界区保护。

问题2：start一个auto reload的timer，会导致8分钟后才响应

解决：1. 在newTimerGenericCommand中去添加first timer的判断，区分是否auto reload

2. 修改newSwitchTimerLists。Tick overflow时保证把非auto reload的timer插入到overflow list。

问题3：厂内测到start一个非auto reload的timer，会导致8分钟后才响应

解决：在newTimerGenericCommand中去添加first timer的判断，不需要区分是否auto reload