



复旦微电子

# *FM33G0XX*

## *低功耗系列MCU*

### *应用笔记*

## **RCHF 精度校准**

---

**AN002**

**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](mailto:xingjie@fmsh.com.cn) TEL: 13916427310

陈钊: [chenzhao@fmsh.com.cn](mailto:chenzhao@fmsh.com.cn) TEL: 18616125501

### 水气热表及智能家居:

朱发旺: [zhufawang@fmsh.com.cn](mailto:zhufawang@fmsh.com.cn) TEL: 17749796664

姜涛: [jiangtao@fmsh.com.cn](mailto:jiangtao@fmsh.com.cn) TEL: 18701992908

### 超高频 900M 及物联网相关:

王晓腾: [wangxiaoteng@fmsh.com.cn](mailto:wangxiaoteng@fmsh.com.cn) TEL: 13585663727

王天纵: [wangtianzong@fmsh.com.cn](mailto: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.....                    | 2  |
| 2.1 背景.....                  | 2  |
| 2.2 原理.....                  | 2  |
| 2.2.1 RCHF 原始温度曲线 .....      | 2  |
| 2.2.2 扩展定时器校准 .....          | 4  |
| 2.2.3 经验曲线校准 .....           | 5  |
| 3 实现方法.....                  | 6  |
| 3.1 扩展定时器校准 .....            | 6  |
| 3.2 经验曲线校准 .....             | 6  |
| 3.3 例程使用说明 .....             | 7  |
| 3.3.1 扩展定时器校准 .....          | 7  |
| 3.3.2 经验曲线校准 .....           | 10 |
| 4 建议的实现步骤.....               | 12 |
| 4.1 扩展定时器校准 .....            | 12 |
| 4.1.1 XTLEF 停振判断 .....       | 12 |
| 4.1.2 定时器初始化 .....           | 13 |
| 4.1.3 开始捕捉校准 .....           | 14 |
| 4.2 经验曲线校准 .....             | 16 |
| 4.2.1 获取温度 .....             | 16 |
| 4.2.2 计算补偿 Trim 值 .....      | 16 |
| 4.2.3 RCHF 误差补偿 .....        | 17 |
| 4.2.4 RCHF 经验曲线补偿完整流程图 ..... | 18 |
| 5 注意事项.....                  | 18 |
| 附录.....                      | 1  |
| 1 寄存器.....                   | 1  |
| 1.1 RCC .....                | 1  |
| 1.2 EXTIM .....              | 19 |
| 1.3 ADC .....                | 33 |



|                              |    |
|------------------------------|----|
| 2 库函数 .....                  | 38 |
| 2.1 RCC 库函数 .....            | 38 |
| 2.2 ETIM 库函数 .....           | 58 |
| 2.3 ADC 库函数 .....            | 81 |
| 版本信息 .....                   | 95 |
| 上海复旦微电子集团股份有限公司销售及服务中心 ..... | 96 |



## 图片目录

|                                     |    |
|-------------------------------------|----|
| 图 2 - 1: 8M RCHF 原始的温度误差曲线.....     | 2  |
| 图 2 - 2: 16M RCHF 原始的温度误差曲线.....    | 3  |
| 图 2 - 3: 24M RCHF 原始的温度误差曲线.....    | 3  |
| 图 2 - 4: 36M RCHF 原始温度误差曲线.....     | 4  |
| 图 2 - 5: ETimer 捕捉时序图 .....         | 5  |
| 图 2 - 6: 经验曲线校准原理图.....             | 6  |
| 图 3 - 1: 扩展定时器校准法示例说明.....          | 7  |
| 图 3 - 2: RCHF 偏差后 64 分频波形 .....     | 7  |
| 图 3 - 3: RCHF 校准后 64 分频波形 .....     | 8  |
| 图 3 - 4: 8M RCHF 扩展定时器校准后误差曲线.....  | 8  |
| 图 3 - 5: 16M RCHF 扩展定时器校准后误差曲线..... | 9  |
| 图 3 - 6: 24M RCHF 扩展定时器校准后误差曲线..... | 9  |
| 图 3 - 7: 36M RCHF 扩展定时器校准后误差曲线..... | 9  |
| 图 3 - 8: 经验曲线校准法示例说明.....           | 10 |
| 图 3 - 9: 8M RCHF 经验曲线校准后误差曲线.....   | 10 |
| 图 3 - 10: 16M RCHF 经验曲线校准后误差曲线..... | 11 |
| 图 3 - 11: 24M RCHF 经验曲线校准后误差曲线..... | 11 |
| 图 3 - 12: 36M RCHF 经验曲线校准后误差曲线..... | 12 |
| 图 4 - 1: 标准校准函数.....                | 12 |
| 图 4 - 2: 停振检测代码示例.....              | 13 |
| 图 4 - 3: 定时器配置代码示例.....             | 14 |
| 图 4 - 4: 计数源分频代码示例.....             | 14 |
| 图 4 - 5: 捕捉流程图.....                 | 15 |
| 图 4 - 6: 温度多次采集示例.....              | 16 |
| 图 4 - 7: RCHF Trim 相关出厂值 .....      | 16 |
| 图 4 - 8: 温度补偿 RCHF 流程 .....         | 17 |
| 图 4 - 9: 写入补偿 Trim 值.....           | 17 |
| 图 4 - 10: 经验曲线校准流程.....             | 18 |

## 1 说明

本文档为 FM33G0xx 系列低功耗 MCU 的应用笔记，用于说明使用扩展定时器和经验曲线校准 RCHF 的原理和方法。FM33G0xx 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

### 2.1 背景

RCHF，高频 RC 振荡器，主要用于系统主时钟，8MHz 运行时 MCU 可达到性能与功耗的平衡。FM33G0xx 的高频 RC 振荡器 RCHF 的典型振荡频率为 8MHz，在 8M，16M，24M，36M 等频率下时全温区（-40~+85℃）精度在±5%以内。如需将全温区精度提高，通常可以用以下两种方式进行校准。

1、扩展定时器校准法。需要芯片外接 32768 晶体并占用一个扩展定时器。扩展定时器校准法采用扩展定时器 EXTIM 捕捉 XTALF 输出的 32768Hz 的信号来实现，扩展定时器校准法可以使 RCHF 在全温区达到±0.5%的精度。

2、经验曲线校准法。如果没有外接 32768 晶振时，可以使用经验曲线校准法进行校准。使用 MCU 内部的 ADC 测温获取当前温度值，再根据经验曲线对 RCHF 进行简单的校准。经验曲线校准可以使 RCHF 工作在 8MHz、16MHz、24MHz、36MHz 频率时，在全温区达到±2%的精度。

### 2.2 原理

#### 2.2.1 RCHF 原始温度曲线

由图可以看出，RCHF 频率越高，在全温差范围内频率的误差越大。

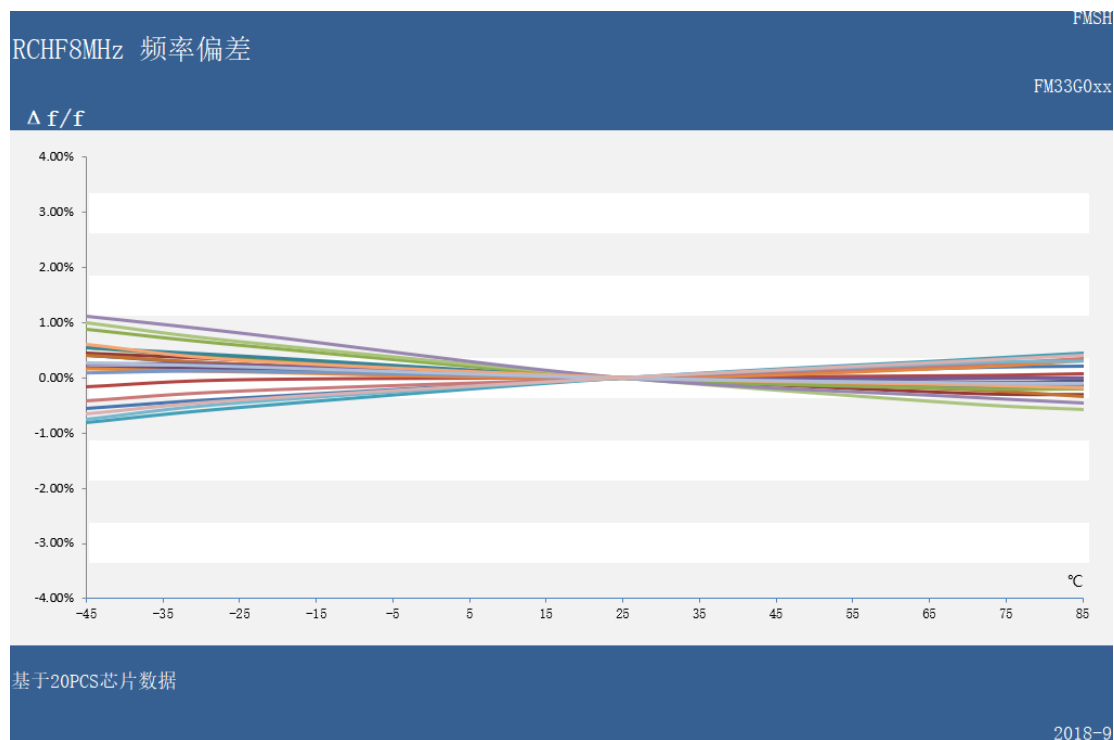


图 2 - 1：8M RCHF 原始的温度误差曲线

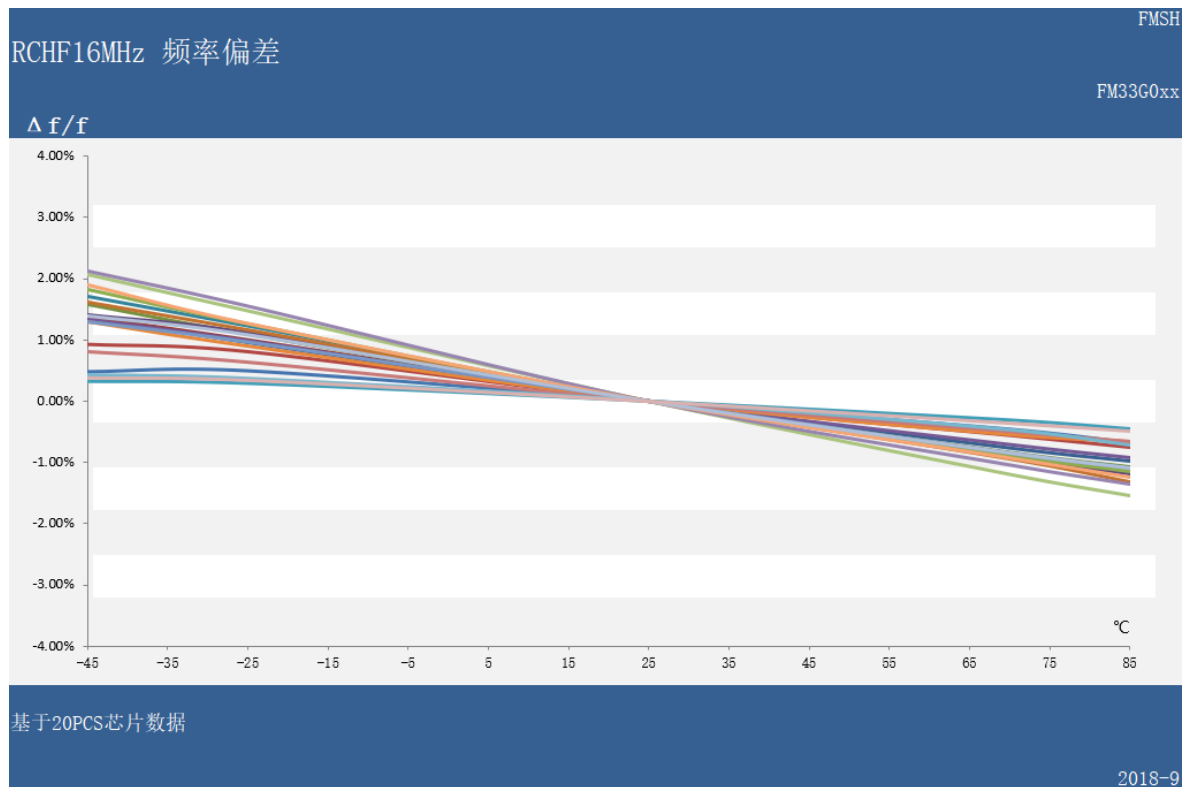


图 2 - 2: 16M RCHF 原始的温度误差曲线

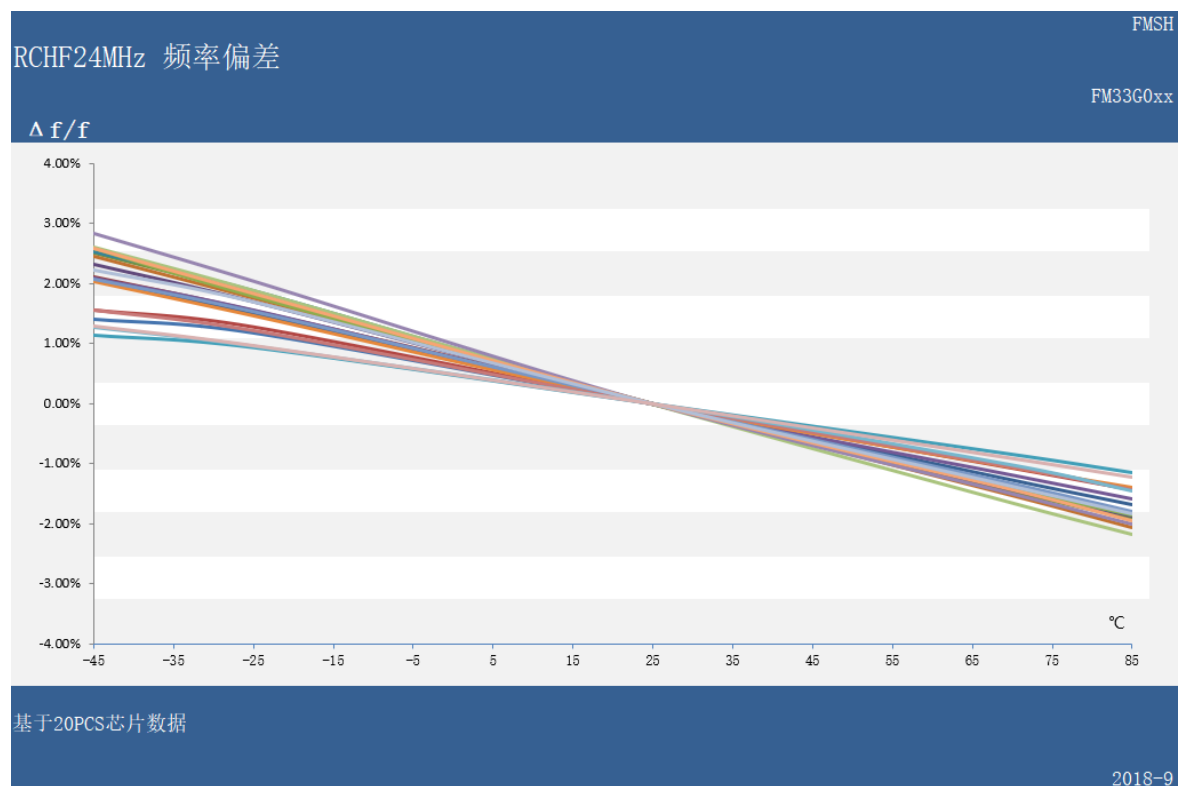


图 2 - 3: 24M RCHF 原始的温度误差曲线

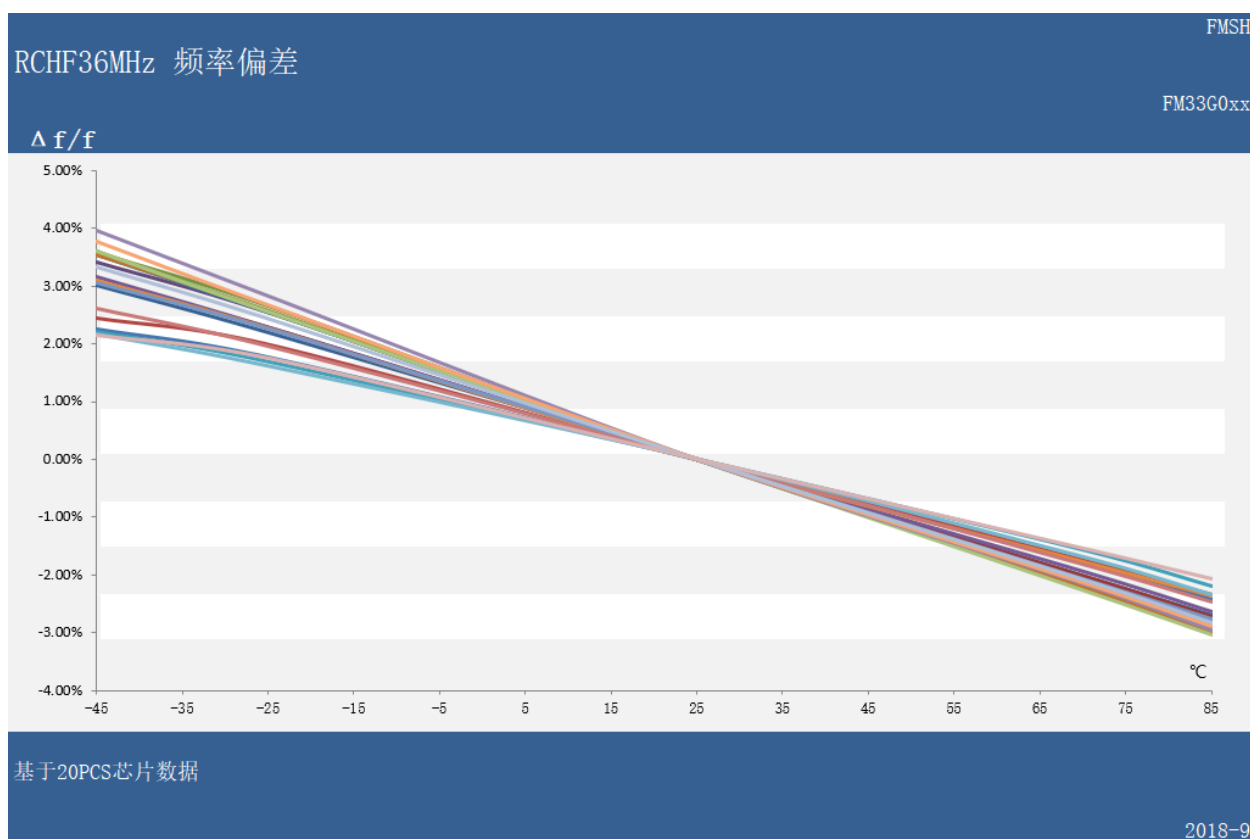


图 2 - 4：36M RCHF 原始温度误差曲线

### 2.2.2 扩展定时器校准

- FM33G0xx 的高频 RC 振荡器支持频率调校，可以使用寄存器 RCHFTRIM 对 RCHF 的频率进行调整，调校范围为中心频率 $\pm 30\%$ ，调校步长为中心频率 0.5%。RCHFTRIM 调校寄存器值范围为 00h-7fh，其中 7'h00 表示频率最低，7'h7F 表示频率最高。
- 低频晶体振荡器 XTLF 为外接 32768Hz 的石英晶体，具有很高的温度稳定性，可作为参考时钟用于校准 RCHF。
- 校准采用 ETimer 的捕捉功能，使用 RCHF(APB 时钟)作为定时器的计数源，对 XTLF 的 128 分频（256Hz，合 3.90625ms）进行脉冲周期捕捉。（ $3.90625\text{ms} = 1 / (32768\text{Hz} / 128)$ ）
- ETimer 为 16 位定时器，8MHz 计数时钟下，在计数器不溢出的情况下可捕捉最长 8.192ms 的脉冲周期，因此捕捉 32768 的 128 分频（256Hz，合 3.90625ms）比较合适，有足够高的精度和余量。（ $8.192\text{ms} = (1/8\text{MHz}) * 65536$ ）
- 使用 ETimer 的带清零的单次捕捉模式进行捕捉，启动捕捉后计数器保持为 0，捕捉到捕捉信号的第一个上升沿后开始计数，捕捉到第二个上升沿后停止计数。

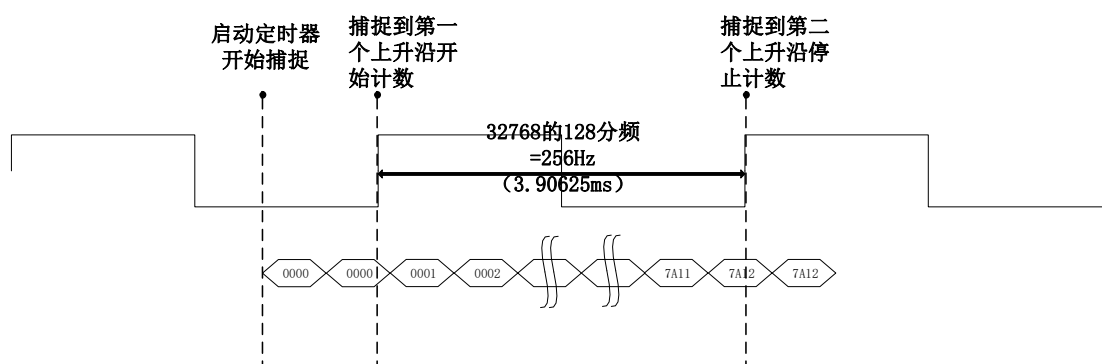


图 2 - 5: ETimer 捕捉时序图

- 理想情况下（RCHF 等于 8MHz 时），定时器对 256Hz 信号的捕捉值等于 31250， $31250 = ((1/256\text{Hz}) / (1/8000000\text{Hz}))$ 。
- 当捕捉值不等于 31250 时，说明 RCHF 的频率偏离了 8MHz。若捕捉值大于 31250 说明 RCHF 频率高于 8MHz，若捕捉值小于 31250 说明 RCHF 频率低于 8MHz。
- RCHFTRIM 寄存器的调校步长为 RCHF 中心频率 0.5%，8MHz 情况下，每个补偿值可调约 40KHz，换算到捕捉值约等于 156.25 个捕捉值（ $156.25 = (40\text{KHz}/8\text{MHz}) * 31250$ ）。
- 当然，RCHF 频率无法校准到理想情况的 0 误差，因此捕捉值进入一定的范围即可认为校准完成，这里我们取  $31250 \pm 120$ ，相当于  $8\text{MHz} \pm 30\text{KHz}$ ，约  $\pm 0.384\%$  的精度（理想情况下，实测校准后全温区精度在  $\pm 0.5\%$  内）。
- 对于 RCHF 不等于 8MHz 的情况，我们在校准时先对计数源进行分频，让计数源等于或者接近 8MHz；16MHz 可以将计数源分到 8MHz；24MHz、36MHz 则可以将计数源分到 6MHz 或 9MHz，接近 8MHz，按比例将捕捉值缩放到 8MHz 处理也可以校准。（主频也可以使用其他频率捕获 XTALF 分频后的脉冲，只需保证捕捉 XTALF 分频后的脉冲有足够高的精度，同时有足够的余量。）

### 2.2.3 经验曲线校准

下图为理论上的误差校准，实际情况根据芯片差异，略有不同。其中蓝色曲线 RCHF 原始温度误差曲线；红色曲线为校准后的温度误差曲线。x 轴为环境温度，y 轴为 RCHF 的频率误差。

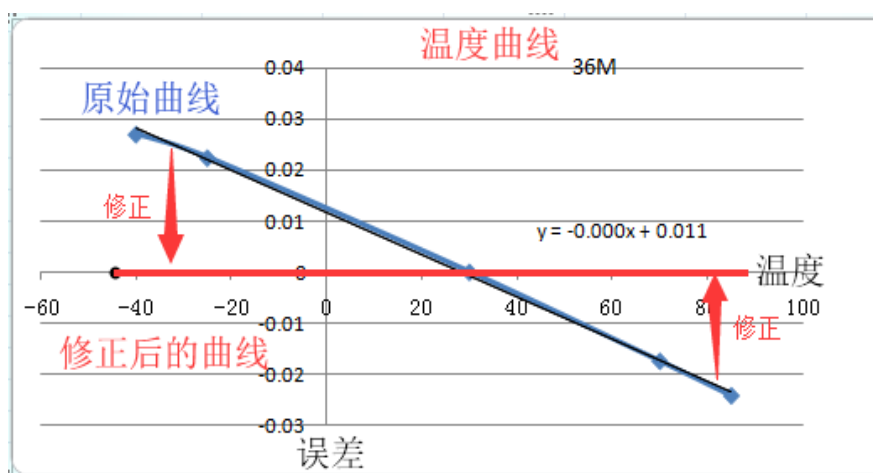


图 2 - 6：经验曲线校准原理图

由数学知识可知， $y = Ax + B$ ，其中 A，B 为 RCHF 频率误差曲线的系数（斜率和截距），该系数通过大量的测试获取，提供给客户使用用于校准 RCHF；x 为温度值；y 为对应的 RCHF 的频率误差。因此，当测量获取到 FM33G0xx 的温度时（可以使用 ADC 采集 FM33G0xx 温度），代入  $y = Ax + B$ ，便可估算出当前温度下 RCHF 的误差，根据该误差对 RCHF 进行校准，从而实现 RCHF 在全温区 2% 的精度。

### 3 实现方法

#### 3.1 扩展定时器校准

使用扩展定时器进行校准时，推荐以下步骤进行实现，操作中涉及到的寄存器与库函数可以查阅本文附录。

- (1) 选择定时器时钟源及计数源；
- (2) 根据系统运行时钟频率，对时钟源及计数源进行分频处理；
- (3) 配置定时器：脉冲周期单次捕获模式；
- (4) 根据捕捉脉冲宽度与理论计算值相比较，修改 RCHF 的 TRIM 值；
- (5) 重复以上步骤，直至 RCHF 精度达到系统需求。

#### 3.2 经验曲线校准

使用经验曲线进行校准时，推荐以下步骤进行实现，操作中涉及到的寄存器与库函数可以查阅本文附录。

- (1) 配置 ADC 转换：内部温度测量；
- (2) 采集温度，并计算该 RCHF 频率下，RCHF 的 TRIM 值；
- (3) 修改 RCHF 的 TRIM 值。

### 3.3 例程使用说明

#### 3.3.1 扩展定时器校准

扩展定时器校准例程请参考 ETIM 校准 RCHF 示例；

| << FM33G0xx示例程序 > ETIM <span style="float: right;">搜索"ETIM"</span> |                |     |    |  |
|--|----------------|-----|----|--|
| 名称   | 修改日期           | 类型  | 大小 |  |
| ETIM的PWM输出   | 2018/9/12 0:54 | 文件夹 |    |  |
| ETIM定时   | 2018/9/12 0:54 | 文件夹 |    |  |
| ETIM级联定时   | 2018/9/12 0:54 | 文件夹 |    |  |
| ETIM输入捕捉   | 2018/9/12 0:54 | 文件夹 |    |  |
| ETIM外部信号计数比较匹配   | 2018/9/12 0:53 | 文件夹 |    |  |
| ETIM校准RCHF   | 2018/9/12 0:54 | 文件夹 |    |  |

图 3 - 1：扩展定时器校准法示例说明

示例程序中，将 RCHF 的 Trim 值故意填错，并将主频 64 分频后再 FOUT (PG6) 管脚输出。通过通过控制校准函数是否调用，借助示波器可以查看到校准前和校准后的 RCHF 64 分频后的波形。

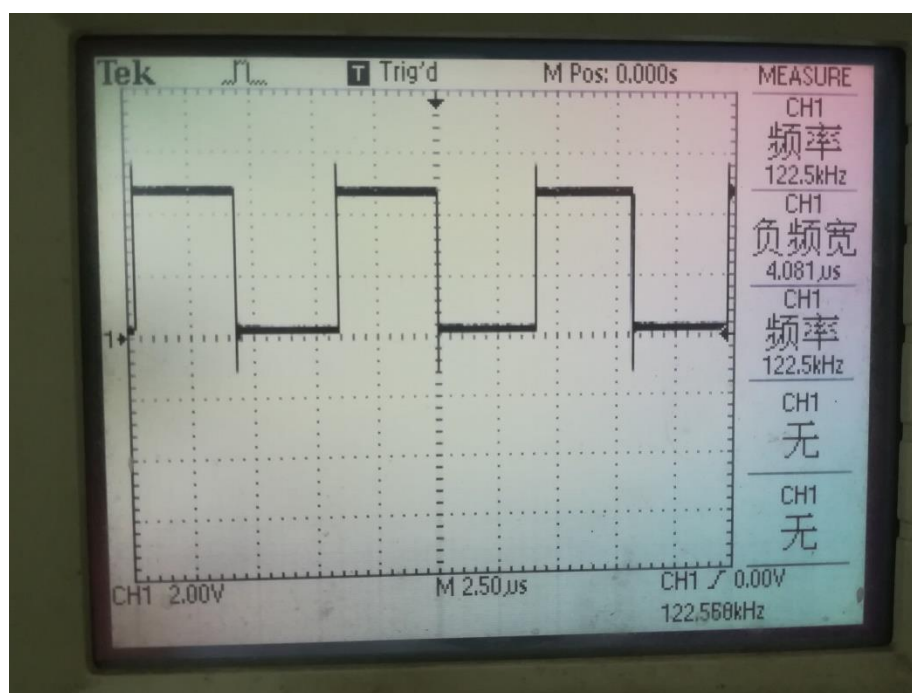


图 3 - 2：RCHF 偏差后 64 分频波形

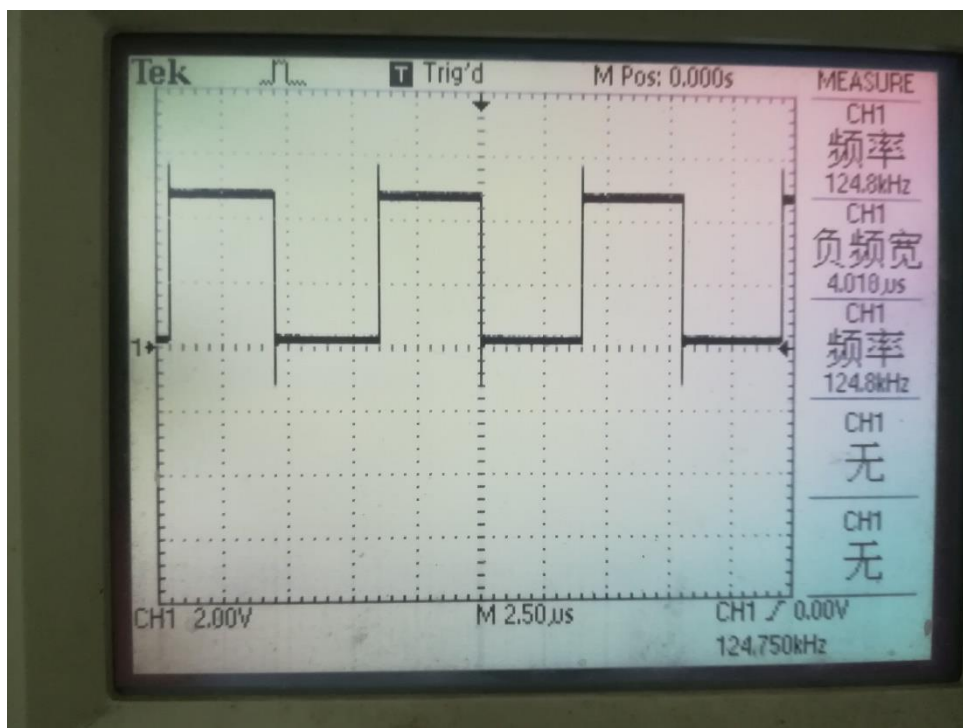


图 3 - 3: RCHF 校准后 64 分频波形

经扩展定时器校准后的测试数据如下。

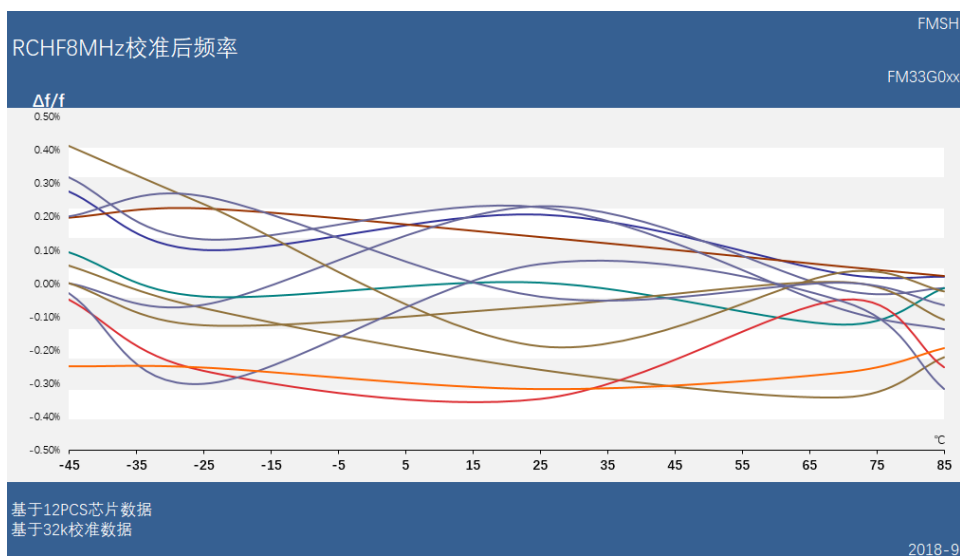


图 3 - 4: 8M RCHF 扩展定时器校准后误差曲线

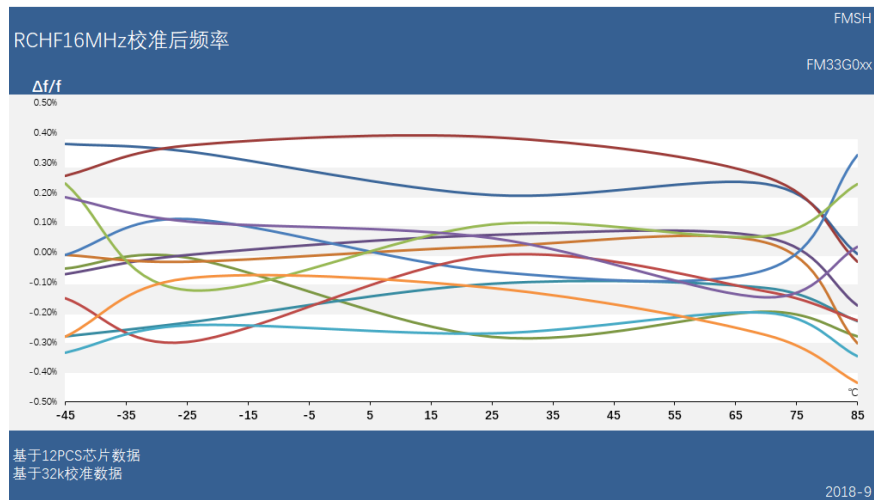


图 3 - 5: 16M RCHF 扩展定时器校准后误差曲线

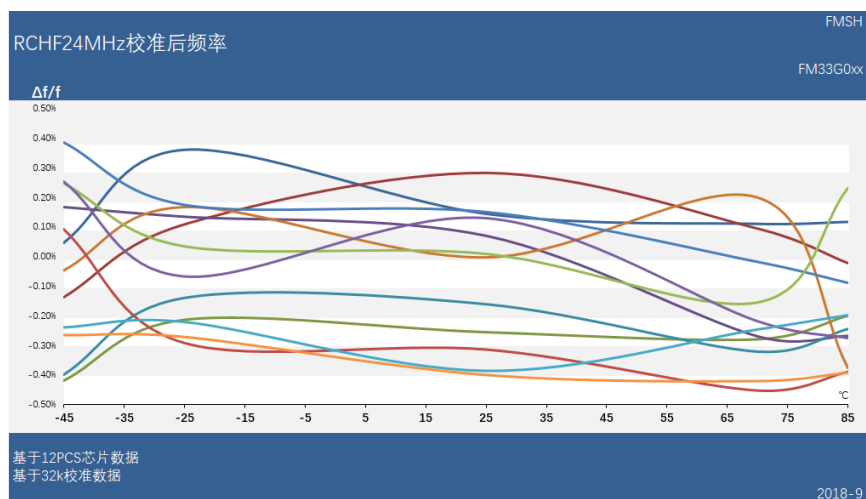


图 3 - 6: 24M RCHF 扩展定时器校准后误差曲线

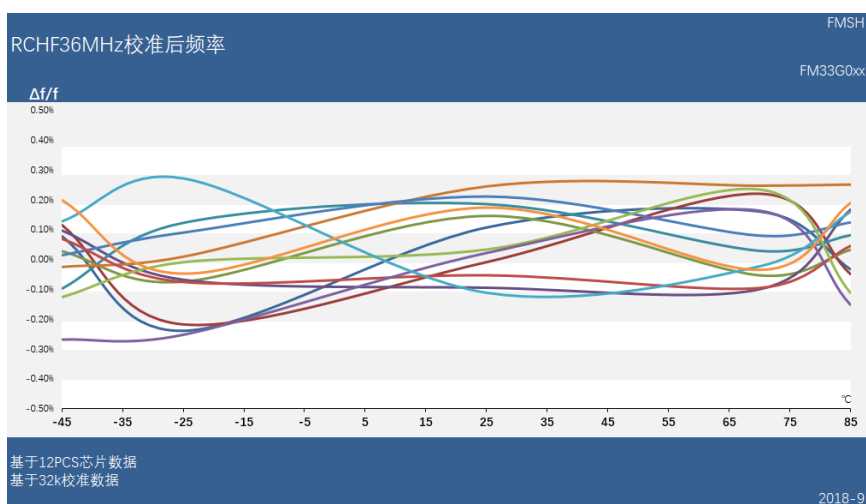


图 3 - 7: 36M RCHF 扩展定时器校准后误差曲线

### 3.3.2 经验曲线校准

经验曲线校准例程请参考 RCHF 根据温度补偿示例。

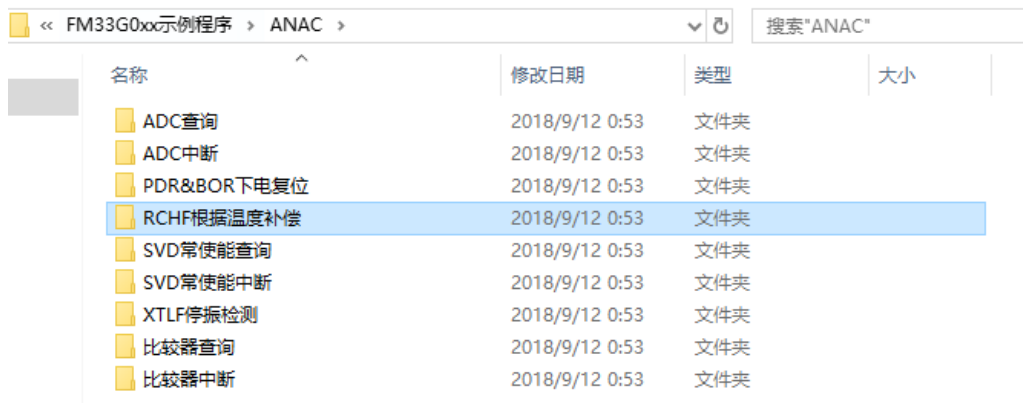


图 3 - 8：经验曲线校准法示例说明

经经验曲线校准后的测试数据如下。

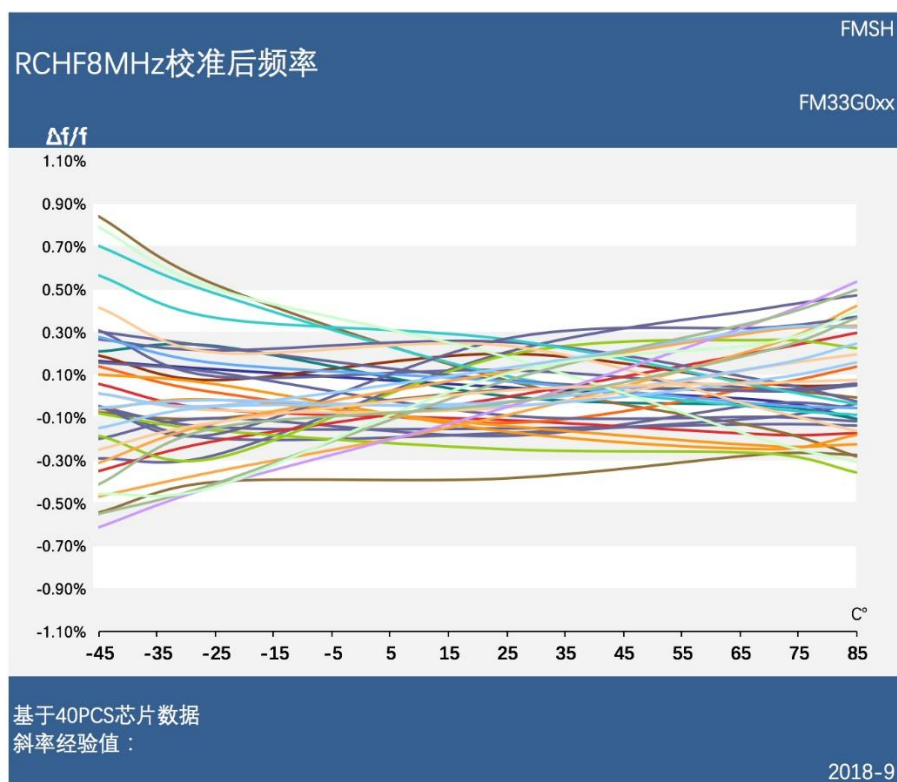


图 3 - 9：8M RCHF 经验曲线校准后误差曲线

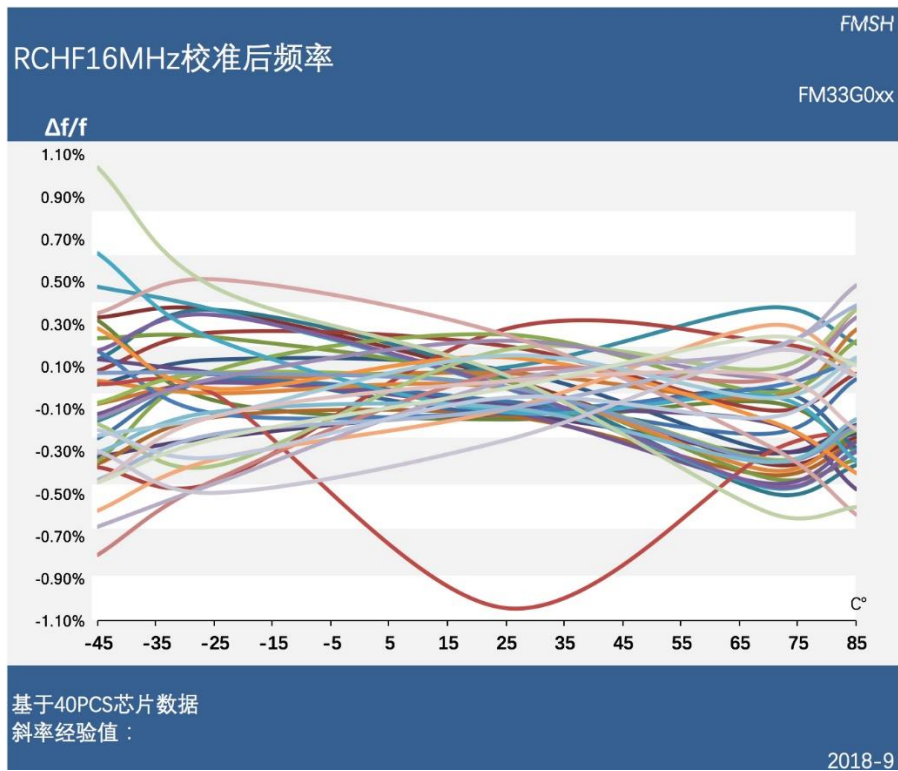


图 3 - 10: 16M RCHF 经验曲线校准后误差曲线

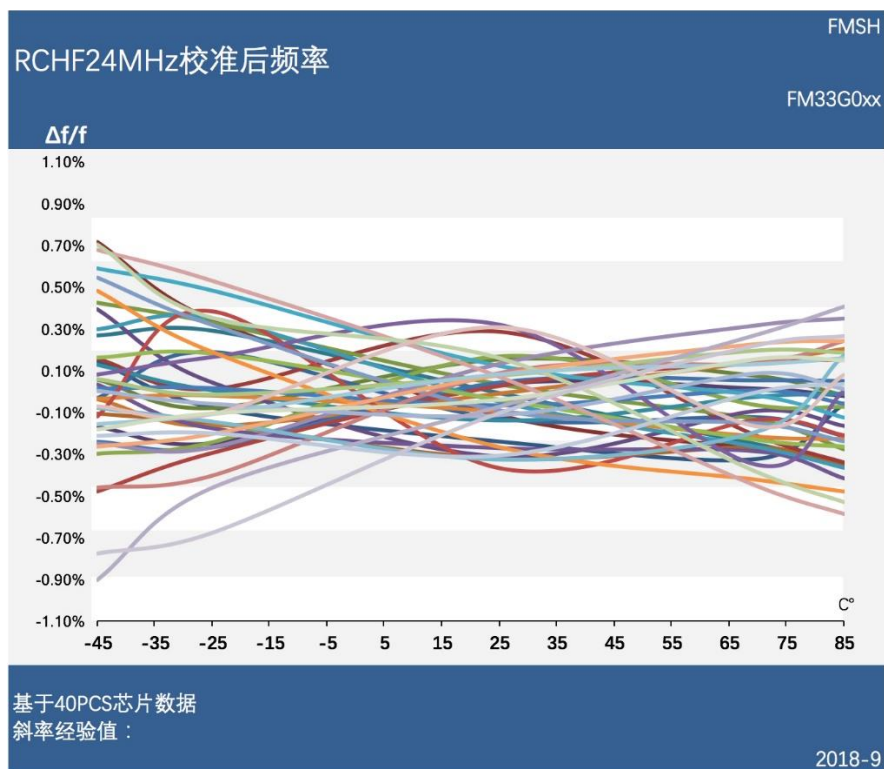


图 3 - 11: 24M RCHF 经验曲线校准后误差曲线

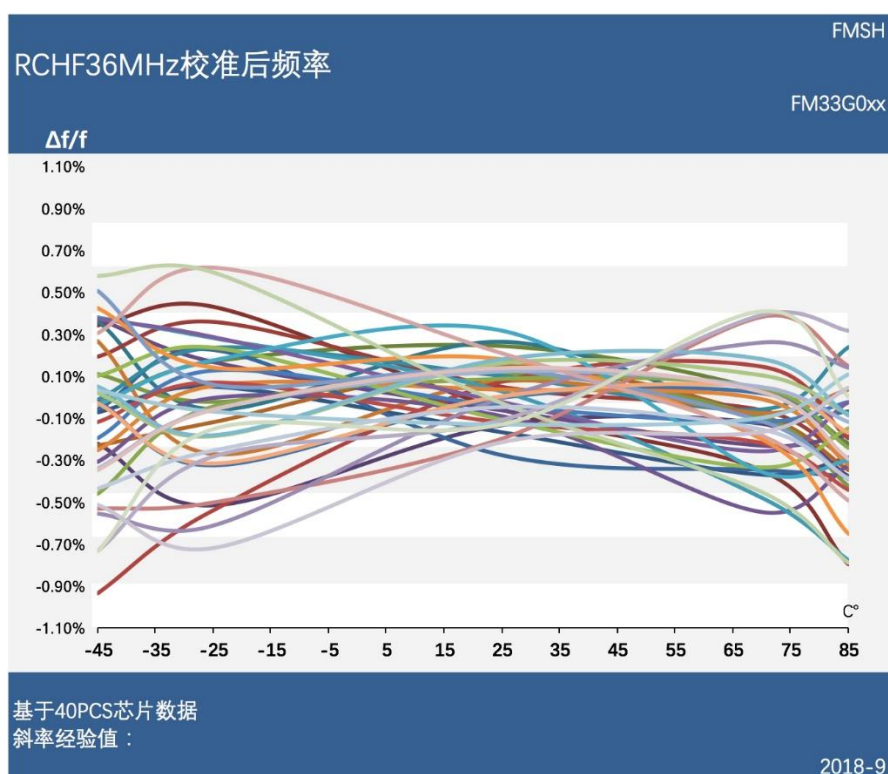


图 3 - 12: 36M RCHF 经验曲线校准后误差曲线

## 4 建议的实现步骤

### 4.1 扩展定时器校准

扩展定时器校准，可以参考示例程序，使用示例程序中的校准函数进行校准，校准函数如下：

```
//xtlf的32768Hz时钟作为参考源  
//本函数仅适用于APB不分频的情况  
//ClkMode 1 = 8M  
//ClkMode 2 = 16M  
//ClkMode 3 = 24M  
//ClkMode 4 = 32M  
void RCHF_Adj_Proc(ETIMx_Type* ETIMx, uint08 ClkMode )
```

图 4 - 1: 标准校准函数

其中，输入参数 1 为扩展定时器名字，调用时可使用任意一个空闲的扩展定时器（ETIM1~ETIM4）；输入参数 2 为系统时钟频率信息，本函数仅适用于 APB 不分频的情况，因此 APBClk=AHBClk。

#### 4.1.1 XTLF 停振判断

开始校准前先判断 XTLF 是否停振，确保 XTLF 正常再启动校准，否则按芯片出厂前测试的常温校准值进行校准。

```
//FDET0 停振检测模块输出
//1: XTLF未停振
//0: XTLF停振
if( RESET == ANAC_FDETIF_FDET0_Chk())//xtlf停振
{
    //使用测试参数
    RCC_Init_RCHF_Trim(ClkMode);
    return;
}
```

图 4 - 2：停振检测代码示例

#### 4.1.2 定时器初始化

启动校准前先对定时器进行初始化配置：

- 配置计数信号和捕捉信号分别为 APBCLK（计数信号），XTLF（捕捉信号）
- 计数源 1 分频（不分频）
- 捕捉源 128 分频
- 引脚输入数字滤波使能打开（不起作用）
- 工作模式选择捕捉模式
- 扩展定时器级联关闭
- 计数模式采沿方式选择上升沿计数
- PWM 输出关闭
- 捕捉模式选择周期捕捉
- 带清零捕捉模式使能
- 单次捕使能
- 捕捉沿选择上升沿捕捉
- 关闭全部定时器中断
- 计数初值清零
- 比较寄存器清零

```

5 void RCHF_Adj_EtimCfg(ETIMx_Type* ETIMx)
6 {
7     ETIM_InitTypeDef init_para;
8     volatile uint08 EtimNum;
9
10    EtimNum = ((uint32_t)ETIMx - ETIMER1_BASE)>>5;
11
12    //信号源参数
13    init_para.sig_src_para.SIG1SEL = ETIMx_ETxINSEL_SIG1SEL_GROUP1; /* 内部信号1源选择GROUP1 */
14    init_para.sig_src_para.SIG2SEL = ETIMx_ETxINSEL_SIG2SEL_GROUP2; /* 内部信号2源选择GROUP2 */
15
16    switch(EtimNum)
17    {
18    case 0:
19        init_para.sig_src_para.PRESCALE1 = 0; /* 1分频 */ /* ETx预分频寄存器1 */ /* 00表示1分频，FF表示256分频 */
20        init_para.sig_src_para.PRESCALE2 = 127; /* 128分频 */ /* ETx预分频寄存器2 */ /* 00表示1分频，FF表示256分频 */
21
22        //控制类参数
23        init_para.ctrl_para.EXFLT = ENABLE; /* 引脚输入数字滤波使能打开 */
24        init_para.ctrl_para.MOD = ETIMx_ETxCR_MOD_CAPTURE; /* 工作模式选择捕捉模式 */
25        init_para.ctrl_para.CASEN = DISABLE; /* 扩展定时器捕获关闭 */
26        init_para.ctrl_para.EDGESEL = ETIMx_ETxCR_EDGESEL_RISING; /* 计数模式采样方式选择 (计数时钟选择mcu_clk时该位无效，总是采用mcu_clk时钟上升沿计数) */
27
28        init_para.ctrl_para.PWM = DISABLE; /* PWM输出关闭 */
29
30        init_para.ctrl_para.CAPMOD = ETIMx_ETxCR_CAPMOD_PERIOD; /* 捕捉模式控制 */
31        init_para.ctrl_para.CAPCLR = ENABLE; /* 清除捕捉模式控制 */ /* 使能后计数器保持0，捕捉到第一个有效沿之后timer才开始计数，之后的捕
32        init_para.ctrl_para.CAPONCE = ENABLE; /* 单次捕捉控制 */
33        init_para.ctrl_para.CAPEDGE = ETIMx_ETxCR_CAPEDGE_RISING; /* 捕捉沿选择 */
34
35        init_para.ctrl_para.CMPTE = DISABLE; /* 扩展定时器比较中断使能 */
36        init_para.ctrl_para.CAPTE = DISABLE; /* 扩展定时器捕获中断使能 */
37        init_para.ctrl_para.OVIE = DISABLE; /* 扩展定时器溢出中断使能 */
38
39        init_para.ctrl_para.INITVALUE = 0; /* 计数初值 */ /* ETx初值寄存器 */
40        init_para.ctrl_para.CMP = 0; /* ETx比较寄存器 */
41
42        init_para.ctrl_para.CEN = DISABLE; /* 启动控制 */
43
44        //初始化ETIM
45        ETIMx_Init(ETIMx, &init_para);
46
47        // ETIMx_ETxCR_CEN_Setable(ETIMx, ENABLE); //启动定时器
48    }
49 }

```

图 4 - 3：定时器配置代码示例

- 根据 RCHF 频率选择计数源分频并确定捕捉值缩放比例

```

switch(ClkMode)
{
    case 1://8
        ETIMx_ETxPESCALE1_Write(ETIMx, 0); //计数源1分频
        DIVClkmode = 1.0;
        break;

    case 2://16/2
        ETIMx_ETxPESCALE1_Write(ETIMx, 1); //计数源2分频
        DIVClkmode = 1.0;
        break;

    case 3://24/4
        ETIMx_ETxPESCALE1_Write(ETIMx, 3); //计数源4分频
        DIVClkmode = 8.0/6.0;
        break;

    case 4://36/4
        ETIMx_ETxPESCALE1_Write(ETIMx, 3); //计数源4分频
        DIVClkmode = 8.0/9.0;
        break;

    default://8
        ETIMx_ETxPESCALE1_Write(ETIMx, 0); //计数源不分频
        DIVClkmode = 1.0;
        break;
}

```

图 4 - 4：计数源分频代码示例

### 4.1.3 开始捕捉校准

捕捉校准流程如下。

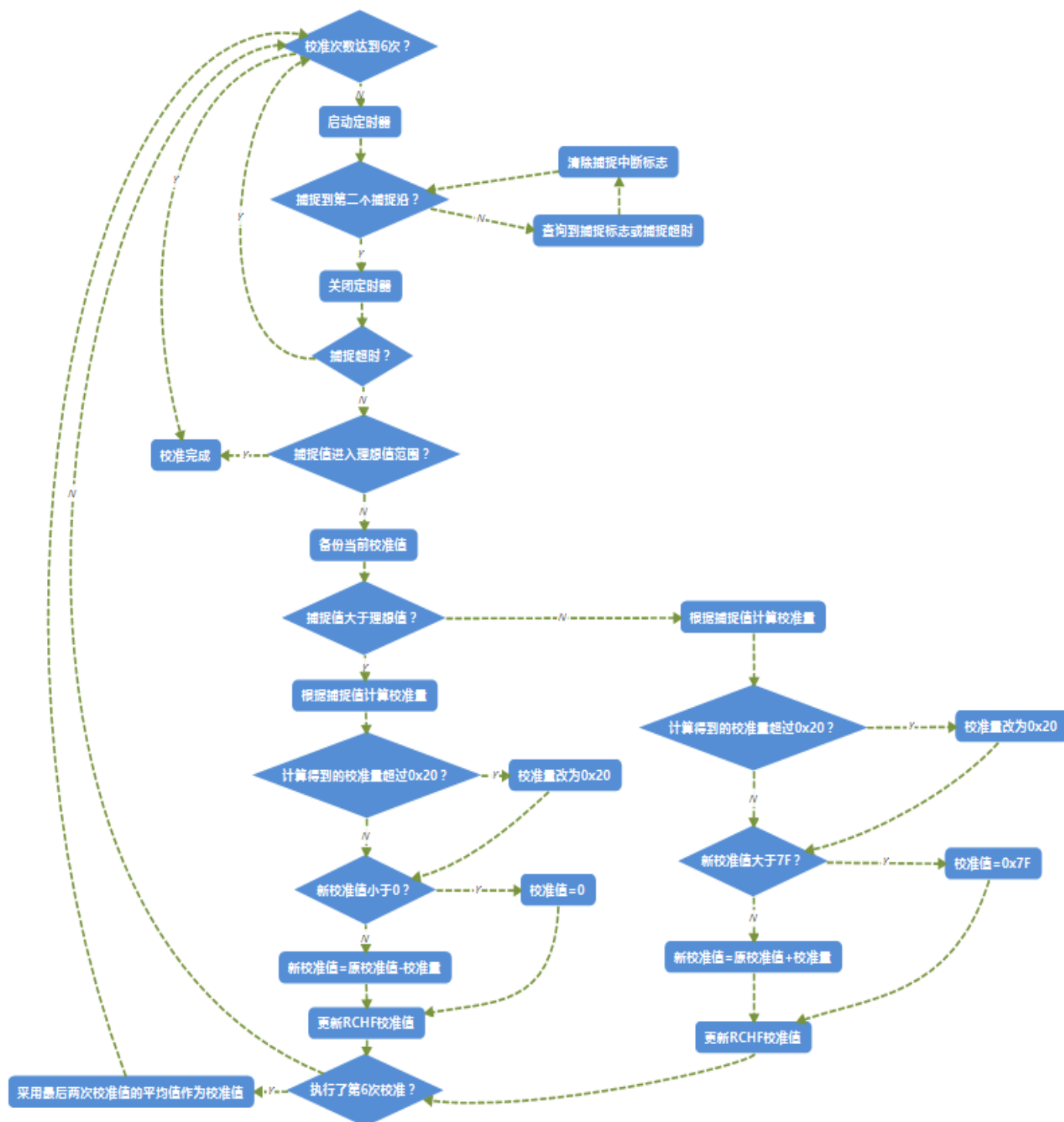


图 4-5: 捕捉流程图

## 4.2 经验曲线校准

相比扩展定时器校准法，经验曲线校准较为简单。在示例程序中，经验曲线校准法使用到了三个方法：获取温度、计算补偿 Trim 值、RCHF 误差修正。

### 4.2.1 获取温度

环境温度，使用 MCU 内部温度传感器获取。ADC 相关配置及温度采集可以参考示例程序。为提高减小的采集温度误差的影响，在温度采集时，连续采集 8 次，取平均值后，作为当前的环境温度，用于 RCHF 校准。

```
for(i=0; i<8; i++)
{
    ANAC_ADCIF_ADC_IF_Clr();           //清除中断标志

    ADCData[i] = 0;
    if(0 == ADC_Wait_Finish())         //等待转换完成
    {
        ADCData[i] = ANAC_ADCDATA_Read(); //读取AD值
        fTempADC += ADCData[i];
    }
    else
    {
        break;
    }
}
```

图 4 - 6：温度多次采集示例

### 4.2.2 计算补偿 Trim 值

补偿 Trim 值计算，需要使用出厂写入 NVM 中的温度曲线相关数据。

```
//RCHF trim值数组
// 8M、16M、24M、36M
#define const_RCHF8_TRIM_MINUS_BUF (0x1FFFFCA0) //trimming value for 8M RCHF trim-1 ~ trim-10
#define const_RCHF8_TRIM_VALUE (0x1FFFFCAA) //trimming value for 8M RCHF trim 0
#define const_RCHF8_TRIM_PLUS_BUF (0x1FFFFCAB) //trimming value for 8M RCHF trim+1 ~ trim+10

#define const_RCHF16_TRIM_MINUS_BUF (0x1FFFFCB5) //trimming value for 16M RCHF trim-1 ~ trim-10
#define const_RCHF16_TRIM_VALUE (0x1FFFFCBF) //trimming value for 16M RCHF trim 0
#define const_RCHF16_TRIM_PLUS_BUF (0x1FFFFCC0) //trimming value for 16M RCHF trim+1 ~ trim+10

#define const_RCHF24_TRIM_MINUS_BUF (0x1FFFFCCA) //trimming value for 24M RCHF trim-1 ~ trim-10
#define const_RCHF24_TRIM_VALUE (0x1FFFFCD4) //trimming value for 24M RCHF trim 0
#define const_RCHF24_TRIM_PLUS_BUF (0x1FFFFCD5) //trimming value for 24M RCHF trim+1 ~ trim+10

#define const_RCHF48_36_TRIM_MINUS_BUF (0x1FFFFDB5) //trimming value for 48_36M RCHF trim-1 ~ trim-10
#define const_RCHF48_36_TRIM_VALUE (0x1FFFFDBF) //trimming value for 48_36M RCHF trim 0
#define const_RCHF48_36_TRIM_PLUS_BUF (0x1FFFFDC0) //trimming value for 48_36M RCHF trim+1 ~ trim+10

//曲线系数 y = Ax + B
#define RCHF8_A -0.0062042564
#define RCHF8_B 3.9769117542

#define RCHF16_A -0.1689410021
#define RCHF16_B 7.8673548504

#define RCHF24_A -0.2793671532
#define RCHF24_B 10.0724720760

#define RCHF48_36_A -0.4117872098
#define RCHF48_36_B 11.5773288576
```

图 4 - 7：RCHF Trim 相关出厂值

曲线系数可以计算获取当前温度下的理想误差，结合 Trim 值数组数据，对 RCHF 的误差进行修正。修正算法，可以自行实现，推荐使用示例程序误差修正算法。

需要注意：对根据温度计算出的 Trim 值进行范围判断，若超出正常范围，应放弃该次补偿。

```
//根据温度补偿rchf
int RCHF_Temperature_Modify(void)
{
    int32_t delta_trim;//RCHF的 补偿trim
    float fTemperature;//当前温度
    fTemperature = ADC_Temperature_Get();//获取当前温度
    delta_trim = RCHF_Temperature_Trim_Get( clkmode ,fTemperature);//各温度下的补偿trim
    //出错判断
    if((delta_trim>20)|| (delta_trim<-20))
    {
        //错误不补偿
        return 1;
    }
    //获取的delta_trim 为温度偏移，还需要常温trim 叠加
    RCC_Init_RCHF_Trim_Modify( clkmode ,delta_trim);//8M trim
    return 0;
}
```

图 4 - 8：温度补偿 RCHF 流程

### 4.2.3 RCHF 误差补偿

计算出 RCHF 补偿 Trim 值后，根据当前 RCHF 频率，将 Trim 写入相应位置即完成使用经验曲线对 RCHF 的校准。注意读改写的顺序。

```
230 //*****
231 rchf常温校准值根据温度修正值修正
232 功能:rchf常温校准值温度修正值修正
233 输入: 要设置的RCHF频率
234         ClkMode 1 = 8M
235         ClkMode 2 = 16M
236         ClkMode 3 = 24M
237         ClkMode 4 = 32M
238         delta_trim 温度修正值
239 *****
240 void RCC_Init_RCHF_Trim_Modify( uint8_t ClkMode ,int32_t delta_trim)
241 {
242     uint32_t Temp32;
243
244     if( ClkMode == 1 ) //8M
245     {
246         Temp32 = const_rchf_Trim8;
247     }
248     else if( ClkMode == 2 )//16M
249     {
250         Temp32 = const_rchf_Trim16;
251     }
252     else if( ClkMode == 3 )//24M
253     {
254         Temp32 = const_rchf_Trim24;
255     }
256     else if( ClkMode == 4 )//36M
257     {
258         Temp32 = const_rchf_Trim36;
259     }
260     else//默认8M
261     {
262         Temp32 = const_rchf_Trim8;
263     }
264
265     if( ((Temp32>>16)&0x0000FFFF) == (~Temp32)&0x0000FFFF) //正反码
266     {
267         RCC_RCHFTRIM_Write((Temp32&0x0000007F)+delta_trim);
268     }
269     else
270     {
271         RCC_RCHFTRIM_Write(0x00000040);
272     }
273 }
```

图 4 - 9：写入补偿 Trim 值

#### 4.2.4 RCHF 经验曲线补偿完整流程图

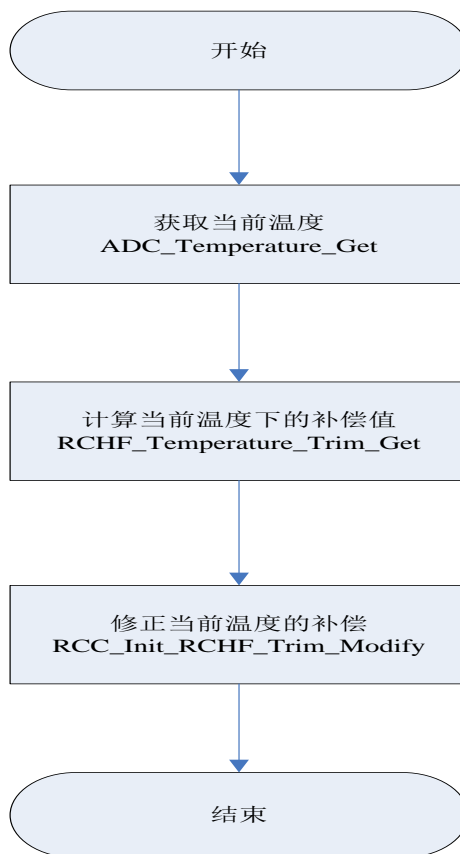


图 4 - 10: 经验曲线校准流程

## 5 注意事项

1、扩展定时器校准时，注意计数源与捕捉源的分频，需保证捕捉的脉冲有足够高的精度，同时计数源要有足够的余量。

2、RCHF 调校时，注意系统主频时钟，根据系统主频时钟进行 RCHF Trim 值的计算。

3、修改 RCHF Trim 值时，注意读改写的顺序。

4、该文档只提供 RCHF 校准的基本方法及思想，对于数据处理的算法，仅供参考，使用者可根据系统需求自行选择其他数据处理算法。推荐直接使用示例程序相关代码。

5、扩展定时器进行 RCHF 校准时，注意计算捕获值，捕获值与系统的捕捉源频率及计数源频率相关。时钟配置若与示例程序不一致时，注意重新计算。



## 附录

### 1 寄存器

#### 1.1 RCC

##### 1.1.1 寄存器

| 地址         | 名称                  | 符号         |
|------------|---------------------|------------|
| 0x40012820 | 停振检测中断使能寄存器         | FDETIE     |
| 0x40012824 | 停振检测中断标志寄存器         | FDETIF     |
| 0x4000020C | 系统时钟配置寄存器           | SYSCLKSEL  |
| 0x40000210 | RCHF 时钟控制寄存器        | RCHFCON    |
| 0x40000214 | RCHF 调校寄存器          | RCHFTRIM   |
| 0x40000218 | PLL 时钟控制寄存器         | PLLCON     |
| 0x4000021C | RCLP 时钟控制寄存器        | RCLPCON    |
| 0x40000220 | RCLP 调校寄存器          | RCLPTRIM   |
| 0x40000224 | XTLF 振荡强度配置寄存器      | XTLFIPW    |
| 0x40000228 | 外设时钟控制寄存器 1         | PERCLKCON1 |
| 0x4000022C | 外设时钟控制寄存器 2         | PERCLKCON2 |
| 0x40000230 | 外设时钟控制寄存器 3         | PERCLKCON3 |
| 0x40000234 | 外设时钟控制寄存器 4         | PERCLKCON4 |
| 0x40000244 | AHB Master 优先级配置寄存器 | MPRIL      |
| 0x40000248 | RCLF 时钟控制寄存器        | RCLFCON    |
| 0x4000024C | RCLF 调校寄存器          | RCLFTRIM   |

##### 1.1.2 停振检测中断使能寄存器

|    |            |
|----|------------|
| 名称 | FDETIE     |
| 地址 | 0x40012820 |



|     |       |       |       |       |       |       |       |         |
|-----|-------|-------|-------|-------|-------|-------|-------|---------|
| 位   | Bit31 | Bit30 | Bit29 | Bit28 | Bit27 | Bit26 | Bit25 | Bit24   |
| 位名  | -     |       |       |       |       |       |       |         |
| 位权限 | U-0   |       |       |       |       |       |       |         |
| 位   | Bit23 | Bit22 | Bit21 | Bit20 | Bit19 | Bit18 | Bit17 | Bit16   |
| 位名  | -     |       |       |       |       |       |       |         |
| 位权限 | U-0   |       |       |       |       |       |       |         |
| 位   | Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8    |
| 位名  | -     |       |       |       |       |       |       |         |
| 位权限 | U-0   |       |       |       |       |       |       |         |
| 位   | Bit7  | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0    |
| 位名  | -     |       |       |       |       |       |       | FDET_IE |
| 位权限 | U-0   |       |       |       |       |       |       | R/W-0   |

| Bit  | 助记符     | 功能描述                                       |
|------|---------|--|
| 31:1 | --      | RFU: 未实现, 读为 0                             |
| 0    | FDET_IE | XTLF 低频检测报警中断使能<br>上电默认关闭, 避免刚上电时晶振未起振触发中断 |

### 1.1.2 停振检测中断标志寄存器

|     |            |       |       |       |       |       |       |       |
|-----|------------|-------|-------|-------|-------|-------|-------|-------|
| 名称  | FDETIF     |       |       |       |       |       |       |       |
| 地址  | 0x40012824 |       |       |       |       |       |       |       |
| 位   | Bit31      | Bit30 | Bit29 | Bit28 | Bit27 | Bit26 | Bit25 | Bit24 |
| 位名  | -          |       |       |       |       |       |       |       |
| 位权限 | U-0        |       |       |       |       |       |       |       |
| 位   | Bit23      | Bit22 | Bit21 | Bit20 | Bit19 | Bit18 | Bit17 | Bit16 |
| 位名  | -          |       |       |       |       |       |       |       |



|     |       |       |       |       |       |       |      |         |
|-----|-------|-------|-------|-------|-------|-------|------|---------|
| 位权限 | U-0   |       |       |       |       |       |      |         |
| 位   | Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9 | Bit8    |
| 位名  | -     |       |       |       |       |       |      |         |
| 位权限 | U-0   |       |       |       |       |       |      |         |
| 位   | Bit7  | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1 | Bit0    |
| 位名  | -     | FDETO | -     |       |       |       |      | FDETIF  |
| 位权限 | U-0   | R-X   | U-0   |       |       |       |      | R/W1C-0 |

| Bit  | 助记符    | 功能描述   |
|------|--------|--|
| 31:7 | --     | RFU: 未实现, 读为 0   |
| 6    | FDETO  | 停振检测模块输出<br>1: XTLEF 未停振<br>0: XTLEF 停振                              |
| 5:1  | --     | RFU: 未实现, 读为 0   |
| 0    | FDETIF | 停振检测中断标志寄存器, XTLEF 停振时硬件异步置位, 软件写 1 清零; 只有在 FDETO 不为 0 的情况下才能够清除此寄存器 |

### 1.1.3 系统时钟配置寄存器

|     |                  |         |       |       |       |             |                |               |
|-----|------------------|---------|-------|-------|-------|-------------|----------------|---------------|
| 名称  | SYSCLKSEL        |         |       |       |       |             |                |               |
| 地址  | 0x4000020C       |         |       |       |       |             |                |               |
| 位   | Bit31            | Bit30   | Bit29 | Bit28 | Bit27 | Bit26       | Bit25          | Bit24         |
| 位名  | LPM_RC<br>LP_OFF | LSCKSEL | -     |       |       | WKUPC<br>LK | SLP_EN<br>EXTI | EXTICK<br>SEL |
| 位权限 | R/W-0            | R/W-0   | U-0   |       |       | R/W-0       | R/W-1          | R/W-0         |
| 位   | Bit23            | Bit22   | Bit21 | Bit20 | Bit19 | Bit18       | Bit17          | Bit16         |
| 位名  | -                |         |       |       |       | APBPRES     |                |               |
| 位权限 | U-0              |         |       |       |       | R/W-000     |                |               |



| 位   | Bit15   | Bit14 | Bit13 | Bit12 | Bit11 | Bit10     | Bit9 | Bit8 |
|-----|---------|-------|-------|-------|-------|-----------|------|------|
| 位名  | -       |       |       |       |       | AHBPRES   |      |      |
| 位权限 | U-10000 |       |       |       |       | R/W-011   |      |      |
| 位   | Bit7    | Bit6  | Bit5  | Bit4  | Bit3  | Bit2      | Bit1 | Bit0 |
| 位名  | -       |       |       |       |       | SYSCLKSEL |      |      |
| 位权限 | U-0     |       |       |       |       | R/W-00    |      |      |

| Bit   | 助记符              | 功能描述  |
|-------|------------------|---|
| 31    | LPM_RCLP_OF<br>F | Sleep/DeepSleep 模式下禁止 RCLP<br>1: 休眠模式下关闭 RCLP<br>0: 休眠模式下保持 RCLP 开启   |
| 30    | LSCKSEL          | 无晶体模式下 LSCLK 时钟源选择, 仅在无外部 32768Hz 晶体的应用场景下有效<br>0: RCLP<br>1: RCLF  |
| 29:27 | --               | RFU: 未实现, 读为 0  |
| 26    | WKUPCLK          | 休眠唤醒后默认时钟频率选择<br>1: RCHF/16, 512KHz<br>0: RCHF 8MHz   |
| 25    | SLP_ENEXTI       | Sleep/DeepSleep 模式下 EXTI 采样设置<br>1: Sleep/DeepSleep 模式下使能外部引脚中断采样 (采样时钟为 LSCLK)<br>0: Sleep/DeepSleep 模式下禁止外部引脚中断采样 (将无法产生 EXTI 中断) |
| 24    | EXTICKSEL        | EXTI 中断采样时钟选择<br>1: 外部引脚中断使用 LSCLK 采样<br>0: 外部引脚中断使用 AHBCLK 采样  |



|       |           |  |
|-------|-----------|--|
|       |           | *建议在关闭所有 EXTI 中断的情况下设置，设置完成后再使能 EXTI 中断  |
| 23:19 | --        | RFU: 未实现，读为 0  |
| 18:16 | APBPRES   | APBCLK 时钟分频选择<br>0xx: 不分频<br>100: 2 分频<br>101: 4 分频<br>110: 8 分频<br>111: 16 分频 |
| 15:11 | --        | RFU: 未实现，读为 0  |
| 10:8  | AHBPRES   | AHBCLK 时钟分频选择<br>0xx: 不分频<br>100: 2 分频<br>101: 4 分频<br>110: 8 分频<br>111: 16 分频 |
| 7:2   | --        | RFU: 未实现，读为 0  |
| 1:0   | SYSCLKSEL | 系统时钟源选择，选择后的时钟将作为 AHBCLK 的输入<br>00: RCHF<br>01: RCLF<br>10: LSCLK<br>11: PLL   |

#### 1.1.4 RCHF 时钟控制寄存器

|    |            |
|----|------------|
| 名称 | RCHFCON    |
| 地址 | 0x40000210 |



|     |       |       |       |       |             |       |       |        |
|-----|-------|-------|-------|-------|-------------|-------|-------|--------|
| 位   | Bit31 | Bit30 | Bit29 | Bit28 | Bit27       | Bit26 | Bit25 | Bit24  |
| 位名  | -     |       |       |       |             |       |       |        |
| 位权限 | U-0   |       |       |       |             |       |       |        |
| 位   | Bit23 | Bit22 | Bit21 | Bit20 | Bit19       | Bit18 | Bit17 | Bit16  |
| 位名  | -     |       |       |       | FSEL        |       |       |        |
| 位权限 | U-0   |       |       |       | R/W/Dy-0000 |       |       |        |
| 位   | Bit15 | Bit14 | Bit13 | Bit12 | Bit11       | Bit10 | Bit9  | Bit8   |
| 位名  | -     |       |       |       |             |       |       |        |
| 位权限 | U-0   |       |       |       |             |       |       |        |
| 位   | Bit7  | Bit6  | Bit5  | Bit4  | Bit3        | Bit2  | Bit1  | Bit0   |
| 位名  | -     |       |       |       |             |       |       | RCHFEN |
| 位权限 | U-0   |       |       |       |             |       |       | R/W-1  |

| Bit   | 助记符    | 功能描述  |
|-------|--------|---|
| 31:20 | --     | RFU: 未实现, 读为 0  |
| 19:16 | FSEL   | RCHF 频率选择寄存器<br>0000: 8MHz<br>0001: 16MHz<br>0010: 24MHz<br>1111: 36MHz<br>其他: 禁止使用 |
| 15:1  | --     | RFU: 未实现, 读为 0  |
| 0     | RCHFEN | RCHF 使能寄存器<br>1: 使能 RCHF<br>0: 关闭 RCHF  |



### 1.1.5 RCHF 调校寄存器

| 名称  | RCHFTRIM   |                |       |       |       |       |       |       |
|-----|------------|----------------|-------|-------|-------|-------|-------|-------|
| 地址  | 0x40000214 |                |       |       |       |       |       |       |
| 位   | Bit31      | Bit30          | Bit29 | Bit28 | Bit27 | Bit26 | Bit25 | Bit24 |
| 位名  | -          |                |       |       |       |       |       |       |
| 位权限 | U-0        |                |       |       |       |       |       |       |
| 位   | Bit23      | Bit22          | Bit21 | Bit20 | Bit19 | Bit18 | Bit17 | Bit16 |
| 位名  | -          |                |       |       |       |       |       |       |
| 位权限 | U-0        |                |       |       |       |       |       |       |
| 位   | Bit15      | Bit14          | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  |
| 位名  | -          |                |       |       |       |       |       |       |
| 位权限 | U-0        |                |       |       |       |       |       |       |
| 位   | Bit7       | Bit6           | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |
| 位名  | -          | RCHFTRIM       |       |       |       |       |       |       |
| 位权限 | U-0        | R/W/Dy-1000000 |       |       |       |       |       |       |

| Bit  | 助记符      | 功能描述   |
|------|----------|--|
| 31:7 | --       | RFU: 未实现, 读为 0   |
| 6:0  | RCHFTRIM | RCHF 频率调校寄存器, 7'h00 表示频率最低, 7'h7F 表示频率最高, 调校范围为中心频率 $\pm 30\%$ , 调校步长为中心频率 0.5% 上电后芯片自动从 Flash 信息区读取 8MHz 调校值并写入此寄存器 |

### 1.1.6 PLL 时钟控制寄存器

| 名称 | PLLCON     |
|----|------------|
| 地址 | 0x40000218 |



|     |              |       |       |       |         |          |            |       |
|-----|--------------|-------|-------|-------|---------|----------|------------|-------|
| 位   | Bit31        | Bit30 | Bit29 | Bit28 | Bit27   | Bit26    | Bit25      | Bit24 |
| 位名  | -            |       |       |       |         |          | PLLDB[9:8] |       |
| 位权限 | U-0          |       |       |       |         |          | R/W-01     |       |
| 位   | Bit23        | Bit22 | Bit21 | Bit20 | Bit19   | Bit18    | Bit17      | Bit16 |
| 位名  | PLLDB[7:0]   |       |       |       |         |          |            |       |
| 位权限 | R/W-11110011 |       |       |       |         |          |            |       |
| 位   | Bit15        | Bit14 | Bit13 | Bit12 | Bit11   | Bit10    | Bit9       | Bit8  |
| 位名  | -            |       |       |       |         |          |            |       |
| 位权限 | U-0          |       |       |       |         |          |            |       |
| 位   | Bit7         | Bit6  | Bit5  | Bit4  | Bit3    | Bit2     | Bit1       | Bit0  |
| 位名  | LOCKED       | -     |       |       | PLLOSEL | PLLINSEL |            | PLLEN |
| 位权限 | R-0          | U-0   |       |       | R/W-0   | R/W-00   |            | R/W-0 |

| Bit   | 助记符     | 功能描述  |
|-------|---------|---|
| 31:26 | --      | RFU: 未实现, 读为 0  |
| 25:16 | PLLDB   | PLL 倍频比<br>输入为 32768Hz, 输出为 16.384MHz, 则 PLLDB=0111110011                     |
| 15:8  | --      | RFU: 未实现, 读为 0  |
| 7     | LOCKED  | PLL 锁定标志<br>1: PLL 已锁定<br>0: PLL 未锁定  |
| 6:4   | --      | RFU: 未实现, 读为 0  |
| 3     | PLLOSEL | PLL 输出选择寄存器<br>0: 选择 PLL 一倍输出作为数字电路内的 PLL 时钟<br>1: 选择 PLL 两倍输出作为数字电路内的 PLL 时钟 |



|     |          |   |
|-----|----------|---|
| 2:1 | PLLINSEL | PLL 输入选择寄存器<br>00/11: XTLP<br>01: RCLP<br>10: RCHF/256          |
| 0   | PLLEN    | PLL 使能寄存器<br>1: 使能 PLL 0: 关闭 PLL<br>注意: 当系统时钟使用 PLL 时, 无法关闭 PLL |

### 1.1.7 RCLP 时钟控制寄存器

| 名称  | RCLPCON    |       |       |       |       |       |       |           |
|-----|------------|-------|-------|-------|-------|-------|-------|-----------|
| 地址  | 0x4000021C |       |       |       |       |       |       |           |
| 位   | Bit31      | Bit30 | Bit29 | Bit28 | Bit27 | Bit26 | Bit25 | Bit24     |
| 位名  | -          |       |       |       |       |       |       |           |
| 位权限 | U-0        |       |       |       |       |       |       |           |
| 位   | Bit23      | Bit22 | Bit21 | Bit20 | Bit19 | Bit18 | Bit17 | Bit16     |
| 位名  | -          |       |       |       |       |       |       |           |
| 位权限 | U-0        |       |       |       |       |       |       |           |
| 位   | Bit15      | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8      |
| 位名  | -          |       |       |       |       |       |       |           |
| 位权限 | U-0        |       |       |       |       |       |       |           |
| 位   | Bit7       | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0      |
| 位名  | -          |       |       |       |       |       |       | RCLP_EN_B |
| 位权限 | U-0        |       |       |       |       |       |       | R/Dy-0    |

| Bit | 助记符 | 功能描述 |
|-----|-----|------|
|-----|-----|------|



|      |           |  |
|------|-----------|--|
| 31:1 | --        | RFU: 未实现, 读为 0                                       |
| 0    | RCLP_EN_B | RCLP 使能寄存器, 硬件自动控制, 软件只读<br>1: RCLP 关闭<br>0: RCLP 开启 |

### 1.1.8 RCLP 调校寄存器

| 名称  | RCLPTRIM   |       |       |       |          |       |       |       |
|-----|------------|-------|-------|-------|----------|-------|-------|-------|
| 地址  | 0x40000220 |       |       |       |          |       |       |       |
| 位   | Bit31      | Bit30 | Bit29 | Bit28 | Bit27    | Bit26 | Bit25 | Bit24 |
| 位名  | -          |       |       |       |          |       |       |       |
| 位权限 | U-0        |       |       |       |          |       |       |       |
| 位   | Bit23      | Bit22 | Bit21 | Bit20 | Bit19    | Bit18 | Bit17 | Bit16 |
| 位名  | -          |       |       |       |          |       |       |       |
| 位权限 | U-0        |       |       |       |          |       |       |       |
| 位   | Bit15      | Bit14 | Bit13 | Bit12 | Bit11    | Bit10 | Bit9  | Bit8  |
| 位名  | -          |       |       |       |          |       |       |       |
| 位权限 | U-0        |       |       |       |          |       |       |       |
| 位   | Bit7       | Bit6  | Bit5  | Bit4  | Bit3     | Bit2  | Bit1  | Bit0  |
| 位名  | -          |       |       |       | RCLPTRIM |       |       |       |
| 位权限 | U-0        |       |       |       | R/W-1000 |       |       |       |

| Bit  | 助记符      | 功能描述                   |
|------|----------|------------------------|
| 31:4 | --       | RFU: 未实现, 读为 0         |
| 3:0  | RCLPTRIM | RCLP 频率调校寄存器, 调校步长约 4% |



## 1.1.9 XTLF 振荡强度配置寄存器

| 名称  | XTLFIPW    |       |       |       |       |         |       |       |
|-----|------------|-------|-------|-------|-------|---------|-------|-------|
| 地址  | 0x40000224 |       |       |       |       |         |       |       |
| 位   | Bit31      | Bit30 | Bit29 | Bit28 | Bit27 | Bit26   | Bit25 | Bit24 |
| 位名  | -          |       |       |       |       |         |       |       |
| 位权限 | U-0        |       |       |       |       |         |       |       |
| 位   | Bit23      | Bit22 | Bit21 | Bit20 | Bit19 | Bit18   | Bit17 | Bit16 |
| 位名  | -          |       |       |       |       |         |       |       |
| 位权限 | U-0        |       |       |       |       |         |       |       |
| 位   | Bit15      | Bit14 | Bit13 | Bit12 | Bit11 | Bit10   | Bit9  | Bit8  |
| 位名  | -          |       |       |       |       |         |       |       |
| 位权限 | U-0        |       |       |       |       |         |       |       |
| 位   | Bit7       | Bit6  | Bit5  | Bit4  | Bit3  | Bit2    | Bit1  | Bit0  |
| 位名  | -          |       |       |       |       | XTLFIPW |       |       |
| 位权限 | U-0        |       |       |       |       | R/W-000 |       |       |

| Bit  | 助记符     | 功能描述  |
|------|---------|---|
| 31:3 | --      | RFU: 未实现, 读为 0  |
| 2:0  | XTLFIPW | XTLF 工作电流选择, 电流越大表示振荡强度越高, 上电复位后使用 000 档位起振, 正常工作时推荐使用 100 或 101 档位<br>000 : 450 nA<br>001 : 400 nA<br>010 : 350 nA<br>011 : 300 nA<br>100 : 250 nA<br>101 : 200 nA<br>110 : 150 nA<br>111 : 100 nA |



## 1.1.10 外设时钟控制寄存器 1

| 名称  | PERCLKCON1 |          |          |          |          |          |          |          |
|-----|------------|----------|----------|----------|----------|----------|----------|----------|
| 地址  | 0x40000228 |          |          |          |          |          |          |          |
| 位   | Bit31      | Bit30    | Bit29    | Bit28    | Bit27    | Bit26    | Bit25    | Bit24    |
| 位名  | DCUCKEN    | -        |          |          |          |          |          |          |
| 位权限 | R/W-1      | U-0      |          |          |          |          |          |          |
| 位   | Bit23      | Bit22    | Bit21    | Bit20    | Bit19    | Bit18    | Bit17    | Bit16    |
| 位名  | -          |          |          |          |          |          |          |          |
| 位权限 | U-0        |          |          |          |          |          |          |          |
| 位   | Bit15      | Bit14    | Bit13    | Bit12    | Bit11    | Bit10    | Bit9     | Bit8     |
| 位名  | -          |          |          |          |          | EXTICKEN |          |          |
| 位权限 | U-0        |          |          |          |          | R/W-000  |          |          |
| 位   | Bit7       | Bit6     | Bit5     | Bit4     | Bit3     | Bit2     | Bit1     | Bit0     |
| 位名  | PDCKKEN    | ANACKKEN | IWDCKKEN | SCUCKEN  | PMUCKEN  | RTCKKEN  | LPTFCKEN | LPTRCKEN |
| 位权限 | R/W-0      | R/W-1    | R/W-0    | R/W/Dy-1 | R/W/Dy-1 | R/W-0    | R/W-0    | R/W-0    |

| Bit   | 助记符      | 功能描述  |
|-------|----------|---|
| 31    | DCUCKEN  | 调试控制单元总线时钟使能，用于开启/关闭仿真调试寄存器时钟，高使能                       |
| 30:11 | --       | RFU：未实现，读为 0  |
| 10:8  | EXTICKEN | EXTI 外部引脚中断采样时钟使能，高使能<br>bit10~8 分别用于控制 EXTI2/1/0 的采样时钟 |
| 7     | PDCKKEN  | I/O 引脚配置单元总线时钟使能，用于开启/关闭 I/O 控制寄存器时钟，高使能                |
| 6     | ANACKKEN | 模拟控制模块总线时钟使能，高使能  |
| 5     | IWDCKKEN | IWDT 总线时钟使能，高使能   |



|   |          |  |
|---|----------|--|
|   |          | 访问 IWDTC 寄存器前必须使能此寄存器                    |
| 4 | SCUCKEN  | 系统配置单元总线时钟使能, 用于开启/关闭系统配置寄存器时钟, 高使能      |
| 3 | PMUCKEN  | 功耗模式管理单元总线时钟使能, 用于开启/关闭功耗模式控制寄存器时钟, 高使能  |
| 2 | RTCKEN   | RTC 总线时钟使能, 高使能; 访问 RTC 寄存器前必须使能此寄存器     |
| 1 | LPTCKEN  | LPTIM 计数时钟使能, 高使能;                       |
| 0 | LPTRCKEN | LPTIM 总线时钟使能, 高使能; 访问 LPTIM 寄存器前必须使能此寄存器 |

### 1.1.11 外设时钟控制寄存器 2

| 名称  | PERCLKCON2   |                 |               |             |             |             |              |            |
|-----|--------------|-----------------|---------------|-------------|-------------|-------------|--------------|------------|
| 地址  | 0x4000022C   |                 |               |             |             |             |              |            |
| 位   | Bit31        | Bit30           | Bit29         | Bit28       | Bit27       | Bit26       | Bit25        | Bit24      |
| 位名  | -            |                 |               |             |             |             |              |            |
| 位权限 | U-0          |                 |               |             |             |             |              |            |
| 位   | Bit23        | Bit22           | Bit21         | Bit20       | Bit19       | Bit18       | Bit17        | Bit16      |
| 位名  | -            |                 |               |             |             | ADCKSEL     |              |            |
| 位权限 | U-0          |                 |               |             |             | R/W-110     |              |            |
| 位   | Bit15        | Bit14           | Bit13         | Bit12       | Bit11       | Bit10       | Bit9         | Bit8       |
| 位名  | -            |                 |               |             |             |             |              | ADCKEN     |
| 位权限 | U-0          |                 |               |             |             |             |              | R/W-0      |
| 位   | Bit7         | Bit6            | Bit5          | Bit4        | Bit3        | Bit2        | Bit1         | Bit0       |
| 位名  | WWDTC<br>KEN | RAMBIS<br>TCKEN | FLSEPCK<br>EN | DMACK<br>EN | LCDCKE<br>N | AESCKE<br>N | TRNGCK<br>EN | CRCKE<br>N |
| 位权限 | R/W-0        | R/W-0           | R/W-0         | R/W-0       | R/W-0       | R/W-0       | R/W-0        | R/W-0      |



| Bit   | 助记符         | 功能描述  |
|-------|-------------|---|
| 31:19 | --          | RFU: 未实现, 读为 0  |
| 18:16 | ADCCKSEL    | ADC 工作时钟选择<br>000: RCLF<br>001: RCHF/32<br>010: RCHF/16<br>011: RCHF/8<br>100: RCHF/16<br>101/110/111: RCLF |
| 15:9  | --          | RFU: 未实现, 读为 0  |
| 8     | ADCCKEN     | ADC 时钟使能, 高使能   |
| 7     | WWDTCEN     | WWDTC 总线时钟使能, 高使能   |
| 6     | RAMBISTCKEN | RAMBIST 时钟使能, 用于开启/关闭 RAM 自测试模块时钟, 高使能  |
| 5     | FLSEPCEN    | FLSC (Flash 擦写控制器) 时钟使能, 高使能  |
| 4     | DMACKEN     | DMA 时钟使能, 高使能   |
| 3     | LCDCKEN     | LCD 时钟使能, 高使能   |
| 2     | AESCKEN     | AES 时钟使能, 高使能   |
| 1     | TRNGCKEN    | RNG 时钟使能, 高使能   |
| 0     | CRCKEN      | CRC 时钟使能, 高使能   |

### 1.1.12 外设时钟控制寄存器 3

| 名称  | PERCLKCON3 |       |       |       |       |       |       |        |
|-----|------------|-------|-------|-------|-------|-------|-------|--------|
| 地址  | 0x40000230 |       |       |       |       |       |       |        |
| 位   | Bit31      | Bit30 | Bit29 | Bit28 | Bit27 | Bit26 | Bit25 | Bit24  |
| 位名  | -          |       |       |       |       |       |       | I2CCKE |
| 位权限 | U-0        |       |       |       |       |       |       | R/W-0  |



| 位   | Bit23        | Bit22      | Bit21     | Bit20     | Bit19     | Bit18     | Bit17      | Bit16      |
|-----|--------------|------------|-----------|-----------|-----------|-----------|------------|------------|
| 位名  | LPUART FCKEN | -          |           |           |           |           | U7816C KE1 | U7816C KE0 |
| 位权限 | R/W-0        | U-0        |           |           |           |           | R/W-0      | R/W-0      |
| 位   | Bit15        | Bit14      | Bit13     | Bit12     | Bit11     | Bit10     | Bit9       | Bit8       |
| 位名  | LPUART CKEN  | UARTC CKEN | UART5C KE | UART4C KE | UART3C KE | UART2C KE | UART1C KE  | UART0C KE  |
| 位权限 | R/W-0        | R/W-0      | R/W-0     | R/W-0     | R/W-0     | R/W-0     | R/W-0      | R/W-0      |
| 位   | Bit7         | Bit6       | Bit5      | Bit4      | Bit3      | Bit2      | Bit1       | Bit0       |
| 位名  | -            |            |           |           |           | HSPICKE   | SPI2CKE    | SPI1CKE    |
| 位权限 | U-0          |            |           |           |           | R/W-0     | R/W-0      | R/W-0      |

| Bit   | 助记符        | 功能描述                   |
|-------|------------|------------------------|
| 31:25 | --         | RFU: 未实现, 读为 0         |
| 24    | I2CCKE     | I2C 时钟使能               |
| 23    | LPUFCKEN   | LPUART 功能时钟使能, 高使能     |
| 22:18 | --         | RFU: 未实现, 读为 0         |
| 17    | U7816CKE1  | U7816-1 时钟使能           |
| 16    | U7816CKE0  | U7816-0 时钟使能           |
| 15    | LPUARTCKEN | LPUART 寄存器总线时钟使能, 高使能  |
| 14    | UARTCCKEN  | UART0~5 共享寄存器时钟使能, 高使能 |
| 13    | UART5CKE   | UART5 时钟使能             |
| 12    | UART4CKE   | UART4 时钟使能             |
| 11    | UART3CKE   | UART3 时钟使能             |
| 10    | UART2CKE   | UART2 时钟使能             |
| 9     | UART1CKE   | UART1 时钟使能             |
| 8     | UART0CKE   | UART0 时钟使能             |
| 7:3   | --         | RFU: 未实现, 读为 0         |
| 2     | HSPICKE    | HSPI (SPI0) 时钟使能       |



|   |         |           |
|---|---------|-----------|
| 1 | SPI2CKE | SPI2 时钟使能 |
| 0 | SPI1CKE | SPI1 时钟使能 |

### 1.1.13 外设时钟控制寄存器 4

| 名称  | PERCLKCON4 |       |        |        |        |        |         |         |
|-----|------------|-------|--------|--------|--------|--------|---------|---------|
| 地址  | 0x40000234 |       |        |        |        |        |         |         |
| 位   | Bit31      | Bit30 | Bit29  | Bit28  | Bit27  | Bit26  | Bit25   | Bit24   |
| 位名  | -          |       |        |        |        |        |         |         |
| 位权限 | U-0        |       |        |        |        |        |         |         |
| 位   | Bit23      | Bit22 | Bit21  | Bit20  | Bit19  | Bit18  | Bit17   | Bit16   |
| 位名  | -          |       |        |        |        |        |         |         |
| 位权限 | U-0        |       |        |        |        |        |         |         |
| 位   | Bit15      | Bit14 | Bit13  | Bit12  | Bit11  | Bit10  | Bit9    | Bit8    |
| 位名  | -          |       |        |        |        |        |         |         |
| 位权限 | U-0        |       |        |        |        |        |         |         |
| 位   | Bit7       | Bit6  | Bit5   | Bit4   | Bit3   | Bit2   | Bit1    | Bit0    |
| 位名  | -          |       | ET4CKE | ET3CKE | ET2CKE | ET1CKE | BT34CKE | BT12CKE |
| 位权限 | U-0        |       | R/W-0  | R/W-0  | R/W-0  | R/W-0  | R/W-0   | R/W-0   |

| Bit  | 助记符    | 功能描述              |
|------|--------|-------------------|
| 31:6 | --     | RFU: 未实现, 读为 0    |
| 5    | ET4CKE | 扩展定时器 4 时钟使能, 高使能 |
| 4    | ET3CKE | 扩展定时器 3 时钟使能, 高使能 |
| 3    | ET2CKE | 扩展定时器 2 时钟使能, 高使能 |
| 2    | ET1CKE | 扩展定时器 1 时钟使能, 高使能 |
| 1    | BT2CKE | 基本定时器 2 时钟使能, 高使能 |



|   |        |                   |
|---|--------|-------------------|
| 0 | BT1CKE | 基本定时器 1 时钟使能, 高使能 |
|---|--------|-------------------|

#### 1.1.14 AHB Master 控制寄存器

| 名称  | MPRIL      |       |       |       |       |       |       |       |
|-----|------------|-------|-------|-------|-------|-------|-------|-------|
| 地址  | 0x40000244 |       |       |       |       |       |       |       |
| 位   | Bit31      | Bit30 | Bit29 | Bit28 | Bit27 | Bit26 | Bit25 | Bit24 |
| 位名  | RFUI       | RFUI  | -     |       |       |       |       |       |
| 位权限 | R/W-1      | R/W-0 | U-0   |       |       |       |       |       |
| 位   | Bit23      | Bit22 | Bit21 | Bit20 | Bit19 | Bit18 | Bit17 | Bit16 |
| 位名  | -          |       |       |       |       |       |       |       |
| 位权限 | U-0        |       |       |       |       |       |       |       |
| 位   | Bit15      | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  |
| 位名  | -          |       |       |       |       |       |       |       |
| 位权限 | U-0        |       |       |       |       |       |       |       |
| 位   | Bit7       | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |
| 位名  | -          |       |       |       |       |       |       | MPRIL |
| 位权限 | U-0        |       |       |       |       |       |       | R/W-0 |

| Bit   | 助记符   | 功能描述  |
|-------|-------|---|
| 31:30 | RFUI  | 保留位   |
| 29:1  | --    | RFU: 未实现, 读为 0                                |
| 0     | MPRIL | AHB Master 优先级配置寄存器<br>0: DMA 优先<br>1: CPU 优先 |



## 1.1.15 RCLF 时钟控制寄存器

| 名称  | RCLFCON    |       |       |       |       |       |       |         |
|-----|------------|-------|-------|-------|-------|-------|-------|---------|
| 地址  | 0x40000248 |       |       |       |       |       |       |         |
| 位   | Bit31      | Bit30 | Bit29 | Bit28 | Bit27 | Bit26 | Bit25 | Bit24   |
| 位名  | -          |       |       |       |       |       |       |         |
| 位权限 | U-0        |       |       |       |       |       |       |         |
| 位   | Bit23      | Bit22 | Bit21 | Bit20 | Bit19 | Bit18 | Bit17 | Bit16   |
| 位名  | -          |       |       |       |       |       |       |         |
| 位权限 | U-0        |       |       |       |       |       |       |         |
| 位   | Bit15      | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8    |
| 位名  | -          |       |       |       |       |       |       |         |
| 位权限 | U-0        |       |       |       |       |       |       |         |
| 位   | Bit7       | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0    |
| 位名  | -          |       |       |       |       |       |       | RCLFENB |
| 位权限 | U-0        |       |       |       |       |       |       | R/W-0   |

| Bit  | 助记符     | 功能描述                                   |
|------|---------|--|
| 31:1 | --      | RFU: 未实现, 读为 0                         |
| 0    | RCLFENB | RCLF 使能寄存器<br>0: 使能 RCLF<br>1: 关闭 RCLF |

## 1.1.16 RCLF 调校寄存器

| 名称 | RCLFTRIM   |
|----|------------|
| 地址 | 0x4000024C |



|     |       |       |          |       |       |       |       |       |
|-----|-------|-------|----------|-------|-------|-------|-------|-------|
| 位   | Bit31 | Bit30 | Bit29    | Bit28 | Bit27 | Bit26 | Bit25 | Bit24 |
| 位名  | -     |       |          |       |       |       |       |       |
| 位权限 | U-0   |       |          |       |       |       |       |       |
| 位   | Bit23 | Bit22 | Bit21    | Bit20 | Bit19 | Bit18 | Bit17 | Bit16 |
| 位名  | -     |       |          |       |       |       |       |       |
| 位权限 | U-0   |       |          |       |       |       |       |       |
| 位   | Bit15 | Bit14 | Bit13    | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  |
| 位名  | -     |       |          |       |       |       |       |       |
| 位权限 | U-0   |       |          |       |       |       |       |       |
| 位   | Bit7  | Bit6  | Bit5     | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |
| 位名  | -     |       | RCLFTRIM |       |       |       |       |       |
| 位权限 | U-0   |       |          |       |       |       |       | R/W-0 |

| Bit  | 助记符      | 功能描述                                   |
|------|----------|--|
| 31:6 | --       | RFU: 未实现, 读为 0                         |
| 5:0  | RCLFTRIM | RCLF 调校寄存器<br>0: 使能 RCLF<br>1: 关闭 RCLF |

## 1.2 EXTIM

### 1.2.1 寄存器

| 地址         | 名称           | 符号           |
|------------|--------------|--------------|
| 0x40013090 | ET1 控制寄存器    | ET1CR        |
| 0x40013094 | ET1 输入源选择寄存器 | ET1INSEL     |
| 0x40013098 | ET1 预分频寄存器 1 | ET1PRESCALE1 |
| 0x4001309C | ET1 预分频寄存器 2 | ET1PRESCALE2 |



|            |              |              |
|------------|--------------|--------------|
| 0x400130A0 | ET1 初值寄存器    | ET1IVR       |
| 0x400130A4 | ET1 比较寄存器    | ET1CMP       |
| 0x400130A8 | ET1 中断使能寄存器  | ET1IE        |
| 0x400130AC | ET1 中断标志寄存器  | ET1IF        |
| 0x400130B0 | ET2 控制寄存器    | ET2CR        |
| 0x400130B4 | ET2 输入源选择寄存器 | ET2INSEL     |
| 0x400130B8 | ET2 预分频寄存器 1 | ET2PRESCALE1 |
| 0x400130BC | ET2 预分频寄存器 2 | ET2PRESCALE2 |
| 0x400130C0 | ET2 初值寄存器    | ET2IVR       |
| 0x400130C4 | ET2 比较寄存器    | ET2CMP       |
| 0x400130C8 | ET2 中断使能寄存器  | ET2IE        |
| 0x400130CC | ET2 中断标志寄存器  | ET2IF        |
| 0x400130D0 | ET3 控制寄存器    | ET3CR        |
| 0x400130D4 | ET3 输入源选择寄存器 | ET3INSEL     |
| 0x400130D8 | ET3 预分频寄存器 1 | ET3PRESCALE1 |
| 0x400130DC | ET3 预分频寄存器 2 | ET3PRESCALE2 |
| 0x400130E0 | ET3 初值寄存器    | ET3IVR       |
| 0x400130E4 | ET3 比较寄存器    | ET3CMP       |
| 0x400130E8 | ET3 中断使能寄存器  | ET3IE        |
| 0x400130EC | ET3 中断标志寄存器  | ET3IF        |
| 0x400130F0 | ET4 控制寄存器    | ET4CR        |
| 0x400130F4 | ET4 输入源选择寄存器 | ET4INSEL     |
| 0x400130F8 | ET4 预分频寄存器 1 | ET4PRESCALE1 |
| 0x400130FC | ET4 预分频寄存器 2 | ET4PRESCALE2 |
| 0x40013100 | ET4 初值寄存器    | ET4IVR       |
| 0x40013104 | ET4 比较寄存器    | ET4CMP       |
| 0x40013108 | ET4 中断使能寄存器  | ET4IE        |



|            |             |        |
|------------|-------------|--------|
| 0x4001310C | ET4 中断标志寄存器 | ET4IF  |
| 0x40013110 | ET1 计数值寄存器  | ET1CNT |
| 0x40013114 | ET2 计数值寄存器  | ET2CNT |
| 0x40013118 | ET3 计数值寄存器  | ET3CNT |
| 0x4001311C | ET4 计数值寄存器  | ET4CNT |

### 1.2.2 ET1/2/3/4 控制寄存器

| 名称  | ET1CR/ET2CR/ET3CR/ET4CR                     |       |       |         |        |        |         |         |
|-----|---|-------|-------|---------|--------|--------|---------|---------|
| 地址  | 0x40013090/0x400130B0/0x400130D0/0x400130F0 |       |       |         |        |        |         |         |
| 位   | Bit31                                       | Bit30 | Bit29 | Bit28   | Bit27  | Bit26  | Bit25   | Bit24   |
| 位名  | -   |       |       |         |        |        |         |         |
| 位权限 | U-0   |       |       |         |        |        |         |         |
| 位   | Bit23                                       | Bit22 | Bit21 | Bit20   | Bit19  | Bit18  | Bit17   | Bit16   |
| 位名  | -   |       |       |         |        |        |         |         |
| 位权限 | U-0   |       |       |         |        |        |         |         |
| 位   | Bit15                                       | Bit14 | Bit13 | Bit12   | Bit11  | Bit10  | Bit9    | Bit8    |
| 位名  | -   |       |       |         |        |        | EXFLT   | PWM     |
| 位权限 | U-0   |       |       |         |        |        | R/W-0   | R/W-0   |
| 位   | Bit7  | Bit6  | Bit5  | Bit4    | Bit3   | Bit2   | Bit1    | Bit0    |
| 位名  | CEN   | MOD   | CASEN | EDGESEL | CAPMOD | CAPCLR | CAPONCE | CAPEDGE |
| 位权限 | R/W-0                                       | R/W-0 | R/W-0 | R/W-0   | R/W-0  | R/W-0  | R/W-0   | R/W-0   |

| 位号   | 位名    | 说明                               |
|------|-------|----------------------------------|
| 31:8 | --    | 未实现：读为 0                         |
| 9    | EXFLT | 外部引脚输入数字滤波使能<br>1 = 打开引脚输入信号数字滤波 |



|   |                |   |
|---|----------------|---|
|   |                | 0 = 关闭引脚输入信号数字滤波  |
| 8 | <b>PWM</b>     | PWM 输出控制<br>1 = PWM 输出使能<br>0 = PWM 输出禁止  |
| 7 | <b>CEN</b>     | 启动控制<br>1 = 启动定时器，在计数器模式下启动时将计数初值加载至计数器和<br>工作寄存器；在捕捉模式下，启动时计数器由零开始自由计数，计<br>数到 0xFFFF 后产生溢出信号然后由零开始重新计数<br>0 = 停止计数器计数   |
| 6 | <b>MOD</b>     | 工作模式选择<br>1 = 捕捉模式<br>0 = 定时/计数模式   |
| 5 | <b>CASEN</b>   | <b>Cascade Enable</b> ，扩展定时器级联使能。仅 ET1CR 和 ET3CR 有<br>此位。<br>1 = ET1 (ET3) 和 ET2 (ET4) 级联成 32bit 定时器<br>0 = 16bit 定时器独立工作 |
| 4 | <b>EDGESEL</b> | 计数模式采沿方式选择<br>(计数时钟选择 mcu_clk 时该位无效，总是采用 mcu_clk 时钟上升沿<br>计数)<br>1 = 计数模式采下降沿<br>0 = 计数模式采上升沿                             |
| 3 | <b>CAPMOD</b>  | 捕捉模式控制<br>1 = 脉宽捕捉<br>0 = 脉冲周期捕捉  |
| 2 | <b>CAPCLR</b>  | 带清零捕捉模式控制<br>1 = 事件触发捕捉：使能后计数器保持 0，捕捉到第一个有效沿之后<br>timer 才开始计数   |



|   |                 |  |
|---|-----------------|--|
|   |                 | 0 = 捕捉不清零，计数器一直自由计数  |
| 1 | <b>CAPONCE</b>  | 单次捕捉控制<br>1 = 单次捕捉有效，在捕捉到一次脉冲周期后计数器停止，若需要再次捕捉需重新启动<br>0 = 连续捕捉 |
| 0 | <b>CAPEEDGE</b> | 捕捉沿选择<br>1 = 周期捕捉模式时下沿捕捉<br>0 = 周期捕捉模式时上沿捕捉                    |

### 1.2.3 ET1/2/3/4 输入源选择寄存器

| 名称  | ET1INSEL/ET2INSEL/ET3INSEL/ET4INSEL         |         |       |              |       |       |              |       |
|-----|---|---------|-------|--------------|-------|-------|--------------|-------|
| 地址  | 0x40013094/0x400130B4/0x400130D4/0x400130F4 |         |       |              |       |       |              |       |
| 位   | Bit31                                       | Bit30   | Bit29 | Bit28        | Bit27 | Bit26 | Bit25        | Bit24 |
| 位名  | -   |         |       |              |       |       |              |       |
| 位权限 | U-0   |         |       |              |       |       |              |       |
| 位   | Bit23                                       | Bit22   | Bit21 | Bit20        | Bit19 | Bit18 | Bit17        | Bit16 |
| 位名  | -   |         |       |              |       |       |              |       |
| 位权限 | U-0   |         |       |              |       |       |              |       |
| 位   | Bit15                                       | Bit14   | Bit13 | Bit12        | Bit11 | Bit10 | Bit9         | Bit8  |
| 位名  | -   |         |       |              |       |       |              |       |
| 位权限 | U-0   |         |       |              |       |       |              |       |
| 位   | Bit7  | Bit6    | Bit5  | Bit4         | Bit3  | Bit2  | Bit1         | Bit0  |
| 位名  | SIG2SEL                                     | SIG1SEL | --    | GRP2SEL[2:0] |       |       | GRP1SEL[1:0] |       |
| 位权限 | R/W-0                                       | R/W-0   | U-0   | R/W-000      |       |       | R/W-00       |       |



| 位号   | 位名             | 说明  |
|------|----------------|---|
| 31:8 | ---            | 未实现，读为 0  |
| 7    | <b>SIG2SEL</b> | 内部信号 2 源选择(捕捉源)<br>1 = 扩展定时器 x 的内部信号 2 选择 Group1<br>0 = 扩展定时器 x 的内部信号 2 选择 Group2 |

| 位号 | 位名             | 说明  |
|----|----------------|---|
| 6  | <b>SIG1SEL</b> | 内部信号 1 源选择（在计数模式下计数源仅由此选择，捕捉模式下计数源）<br>1 = 扩展定时器 x 的内部信号 1 选择 Group2<br>0 = 扩展定时器 x 的内部信号 1 选择 Group1 |
| 5  | ---            | 未实现，读为 0  |



|     |              |  |
|-----|--------------|--|
| 4:2 | GRP2SEL[2:0] | <p><b>GROUP2 信号选择控制</b></p> <p><b>ET1</b><br/> 000 = RCLF<br/> 001 = UART1_RX<br/> 010 = XTLF<br/> 011 = ET1_IN1<br/> 100 = ET1_IN2<br/> 101 = CMP1O(比较器 1 输出)<br/> 110 = CMP2O(比较器 2 输出)<br/> 111 = LPTO</p> <p><b>ET2</b><br/> 000 = UART2_RX<br/> 001 = UART3_RX<br/> 010 = XTLF<br/> 011 = ET2_IN1<br/> 100 = ET2_IN2<br/> 101 = CMP1O(比较器 1 输出)<br/> 110 = CMP2O(比较器 2 输出)<br/> 111 = LPTO</p> <p><b>ET3</b><br/> 000 = ET3_IN1<br/> 001 = XTLF<br/> 010 = UART4_RX<br/> 011 = UART5_RX<br/> 100 = RTCSEC<br/> 101~111 = RFU</p> <p><b>ET4</b><br/> 000 = ET4_IN1<br/> 001 = XTLF<br/> 010 = UART_RX2<br/> 011 = UART_RX0<br/> 101 = CMP1O(比较器 1 输出)<br/> 110 = CMP2O(比较器 2 输出)<br/> 110 = RTCSEC<br/> 111 = LPTO</p> |
|-----|--------------|--|



|     |              |   |
|-----|--------------|---|
| 1:0 | GRP1SEL[1:0] | <b>GROUP1 信号选择控制</b><br><b>ET1</b><br>00= APBCLK<br>01= XTLP<br>10= RCLP<br>11= ET1_IN0<br><b>ET2</b><br>00= APBCLK<br>01= XTLP<br>10= RCLP<br>11= ET2_IN0<br><b>ET3</b><br>00= APBCLK<br>01= ET3_IN0<br>10= RTCSEC<br>11= RCLP<br><b>ET4</b><br>00= APBCLK<br>01= ET4_IN0<br>10= RTC64HZ<br>11= LPTO |
|-----|--------------|---|

### 1.2.4 ET1/2/3/4 预分频寄存器 1

| 名称  | ET1PESCALE1/ET2PESCALE1/ET3PESCALE1/ET4PESCALE1 |       |       |       |       |       |       |       |
|-----|---|-------|-------|-------|-------|-------|-------|-------|
| 地址  | 0x40013098/0x400130B8/0x400130D8/0x400130F8     |       |       |       |       |       |       |       |
| 位   | Bit31   | Bit30 | Bit29 | Bit28 | Bit27 | Bit26 | Bit25 | Bit24 |
| 位名  | -   |       |       |       |       |       |       |       |
| 位权限 | U-0   |       |       |       |       |       |       |       |
| 位   | Bit23   | Bit22 | Bit21 | Bit20 | Bit19 | Bit18 | Bit17 | Bit16 |
| 位名  | -   |       |       |       |       |       |       |       |
| 位权限 | U-0   |       |       |       |       |       |       |       |
| 位   | Bit15   | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  |
| 位名  | -   |       |       |       |       |       |       |       |



|     |              |      |      |      |      |      |      |      |
|-----|--------------|------|------|------|------|------|------|------|
| 位权限 | U-0          |      |      |      |      |      |      |      |
| 位   | Bit7         | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
| 位名  | PRESCALE1    |      |      |      |      |      |      |      |
| 位权限 | R/W-00000000 |      |      |      |      |      |      |      |

| 位号   | 位名               | 说明   |
|------|------------------|--|
| 31:8 | ---              | 未实现，读为 0   |
| 7:0  | <b>PRESCALE1</b> | 输入 <b>Signal1</b> (计数源) 的预分频寄存器<br>00 表示 1 分频，FF 表示 256 分频 |

### 1.2.5 ET1/2/3/4 预分频寄存器 2

|     |  |       |       |       |       |       |       |       |
|-----|--|-------|-------|-------|-------|-------|-------|-------|
| 名称  | <b>ET1PESCALE2/ET2PESCALE2/ET3PESCALE2/ET4PESCALE2</b> |       |       |       |       |       |       |       |
| 地址  | 0x4001309C/0x400130BC/0x400130DC/0x400130FC            |       |       |       |       |       |       |       |
| 位   | Bit31  | Bit30 | Bit29 | Bit28 | Bit27 | Bit26 | Bit25 | Bit24 |
| 位名  | -  |       |       |       |       |       |       |       |
| 位权限 | U-0  |       |       |       |       |       |       |       |
| 位   | Bit23  | Bit22 | Bit21 | Bit20 | Bit19 | Bit18 | Bit17 | Bit16 |
| 位名  | -  |       |       |       |       |       |       |       |
| 位权限 | U-0  |       |       |       |       |       |       |       |
| 位   | Bit15  | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  |
| 位名  | -  |       |       |       |       |       |       |       |
| 位权限 | U-0  |       |       |       |       |       |       |       |
| 位   | Bit7   | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |
| 位名  | PRESCALE2  |       |       |       |       |       |       |       |
| 位权限 | R/W-00000000   |       |       |       |       |       |       |       |



| 位号   | 位名               | 说明  |
|------|------------------|---|
| 31:8 | ---              | 未实现，读为 0  |
| 7:0  | <b>PRESCALE2</b> | 输入 <b>Signal2</b> (捕捉源) 的预分频寄存器<br>00 表示 1 分频，FF 表示 256 分频。 |

### 1.2.6 ET1/2/3/4 初值寄存器

| 名称  | ET1IVR/ET2IVR/ET3IVR/ET4IVR                 |       |       |       |       |       |       |       |
|-----|---|-------|-------|-------|-------|-------|-------|-------|
| 地址  | 0x400130A0/0x400130C0/0x400130E0/0x40013100 |       |       |       |       |       |       |       |
| 位   | Bit31                                       | Bit30 | Bit29 | Bit28 | Bit27 | Bit26 | Bit25 | Bit24 |
| 位名  | -   |       |       |       |       |       |       |       |
| 位权限 | U-0   |       |       |       |       |       |       |       |
| 位   | Bit23                                       | Bit22 | Bit21 | Bit20 | Bit19 | Bit18 | Bit17 | Bit16 |
| 位名  | -   |       |       |       |       |       |       |       |
| 位权限 | U-0   |       |       |       |       |       |       |       |
| 位   | Bit15                                       | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  |
| 位名  | INITVALUE[15:8]                             |       |       |       |       |       |       |       |
| 位权限 | R/W-00000000                                |       |       |       |       |       |       |       |
| 位   | Bit7  | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |
| 位名  | INITVALUE[7:0]                              |       |       |       |       |       |       |       |
| 位权限 | R/W-00000000                                |       |       |       |       |       |       |       |

| 位号    | 位名               | 说明         |
|-------|------------------|------------|
| 31:16 | ---              | 未实现，读为 0   |
| 15:0  | <b>INITVALUE</b> | 扩展定时器初值寄存器 |

## 1.2.7 ET1/2/3/4 比较寄存器

| 名称  | ET1CMP/ET2CMP/ET3CMP/ET4CMP                 |       |       |       |       |       |       |       |
|-----|---|-------|-------|-------|-------|-------|-------|-------|
| 地址  | 0x400130A4/0x400130C4/0x400130E4/0x40013104 |       |       |       |       |       |       |       |
| 位   | Bit31                                       | Bit30 | Bit29 | Bit28 | Bit27 | Bit26 | Bit25 | Bit24 |
| 位名  | -   |       |       |       |       |       |       |       |
| 位权限 | U-0   |       |       |       |       |       |       |       |
| 位   | Bit23                                       | Bit22 | Bit21 | Bit20 | Bit19 | Bit18 | Bit17 | Bit16 |
| 位名  | -   |       |       |       |       |       |       |       |
| 位权限 | U-0   |       |       |       |       |       |       |       |
| 位   | Bit15                                       | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  |
| 位名  | CMP[15:8]                                   |       |       |       |       |       |       |       |
| 位权限 | R/W-00000000                                |       |       |       |       |       |       |       |
| 位   | Bit7  | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |
| 位名  | CMP[7:0]                                    |       |       |       |       |       |       |       |
| 位权限 | R/W-00000000                                |       |       |       |       |       |       |       |

| 位号    | 位名  | 说明  |
|-------|-----|---|
| 31:16 | --- | 未实现，读为 0  |
| 15:0  | CMP | 扩展定时器比较寄存器该寄存器与计数器比较，若计数值大于等于该寄存器的值，则产生计数匹配信号至输出控制模块，并产生相应中断。 |

## 1.2.8 ET1/2/3/4 中断使能寄存器

| 名称 | ET1IE/ET2IE/ET3IE/ET4IE |
|----|-------------------------|
|----|-------------------------|



|     |   |       |       |       |       |       |       |       |
|-----|---|-------|-------|-------|-------|-------|-------|-------|
| 地址  | 0x400130A8/0x400130C8/0x400130E8/0x40013108 |       |       |       |       |       |       |       |
| 位   | Bit31                                       | Bit30 | Bit29 | Bit28 | Bit27 | Bit26 | Bit25 | Bit24 |
| 位名  | -   |       |       |       |       |       |       |       |
| 位权限 | U-0   |       |       |       |       |       |       |       |
| 位   | Bit23                                       | Bit22 | Bit21 | Bit20 | Bit19 | Bit18 | Bit17 | Bit16 |
| 位名  | -   |       |       |       |       |       |       |       |
| 位权限 | U-0   |       |       |       |       |       |       |       |
| 位   | Bit15                                       | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  |
| 位名  | -   |       |       |       |       |       |       |       |
| 位权限 | U-0   |       |       |       |       |       |       |       |
| 位   | Bit7  | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |
| 位名  | ---   |       |       |       |       | CMPIE | CAPIE | OVIE  |
| 位权限 | U-0   |       |       |       |       | R/W-0 | R/W-0 | R/W-0 |

| 位号   | 位名    | 说明                                 |
|------|-------|------------------------------------|
| 31:3 | ---   | 未实现：读为 0                           |
| 2    | CMPIE | 扩展定时器比较使能<br>1 = 使能<br>0 = 禁止      |
| 1    | CAPIE | 扩展定时器捕捉中断使能<br>1 = 使能<br>0 = 禁止    |
| 0    | OVIE  | 扩展定时器 x 溢出中断使能<br>1 = 使能<br>0 = 禁止 |



## 1.2.9 ET1/2/3/4 中断标志寄存器

| 名称  | ET1IF/ET2IF/ET3IF/ET4IF                     |       |       |       |         |             |         |         |
|-----|---|-------|-------|-------|---------|-------------|---------|---------|
| 地址  | 0x400130AC/0x400130CC/0x400130EC/0x4001310C |       |       |       |         |             |         |         |
| 位   | Bit31                                       | Bit30 | Bit29 | Bit28 | Bit27   | Bit26       | Bit25   | Bit24   |
| 位名  | -   |       |       |       |         |             |         |         |
| 位权限 | U-0   |       |       |       |         |             |         |         |
| 位   | Bit23                                       | Bit22 | Bit21 | Bit20 | Bit19   | Bit18       | Bit17   | Bit16   |
| 位名  | -   |       |       |       |         |             |         |         |
| 位权限 | U-0   |       |       |       |         |             |         |         |
| 位   | Bit15                                       | Bit14 | Bit13 | Bit12 | Bit11   | Bit10       | Bit9    | Bit8    |
| 位名  | -   |       |       |       |         |             |         |         |
| 位权限 | U-0   |       |       |       |         |             |         |         |
| 位   | Bit7  | Bit6  | Bit5  | Bit4  | Bit3    | Bit2        | Bit1    | Bit0    |
| 位名  | ---   |       |       |       | CMPIF   | EDGEST<br>A | CAPIF   | OVIF    |
| 位权限 | U-0   |       |       |       | R/W1C-0 | R/W1C-0     | R/W1C-0 | R/W1C-0 |

| 位号   | 位名      | 说明  |
|------|---------|---|
| 31:4 | ---     | 未实现：读为 0  |
| 3    | CMPIF   | 比较状态，触发时，硬件置位，软件写 1 清零。<br>1 = 当前计数器的值大于等于比较寄存器的值<br>0 = 当前计数器的值小于比较寄存器的值 |
| 2    | EDGESTA | 捕捉沿状态，触发时，硬件置位，软件写 1 清零。<br>1 = 脉冲宽度捕捉模式时表示捕捉到下沿<br>0 = 脉冲宽度捕捉模式时表示捕捉到上沿  |
| 1    | CAPIF   | 扩展定时器捕捉产生信号，触发时，硬件置位，软件写 1 清零。<br>1 = 捕捉到指定的沿                             |



|   |      |  |
|---|------|--|
|   |      | 0 = 未捕捉到指定的沿   |
| 0 | OVIF | 扩展定时器 x 溢出信号，当计数器的值由 0xFFFF 再增加时将置位，软件写 1 清零。<br>1 = 产生计数溢出<br>0 = 未产生溢出 |

### 1.2.10 ET1/2/3/4 计数值寄存器

| 名称  | ET1CNT/ ET2CNT/ ET3CNT/ ET4CNT              |       |       |       |       |       |       |       |
|-----|---|-------|-------|-------|-------|-------|-------|-------|
| 地址  | 0x40013110/0x40013114/0x40013118/0x4001311C |       |       |       |       |       |       |       |
| 位   | Bit31                                       | Bit30 | Bit29 | Bit28 | Bit27 | Bit26 | Bit25 | Bit24 |
| 位名  | -   |       |       |       |       |       |       |       |
| 位权限 | U-0   |       |       |       |       |       |       |       |
| 位   | Bit23                                       | Bit22 | Bit21 | Bit20 | Bit19 | Bit18 | Bit17 | Bit16 |
| 位名  | -   |       |       |       |       |       |       |       |
| 位权限 | U-0   |       |       |       |       |       |       |       |
| 位   | Bit15                                       | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  |
| 位名  | ETxCNT[15:8]                                |       |       |       |       |       |       |       |
| 位权限 | R   |       |       |       |       |       |       |       |
| 位   | Bit7  | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |
| 位名  | ETxCNT[7:0]                                 |       |       |       |       |       |       |       |
| 位权限 | R   |       |       |       |       |       |       |       |

| 位号    | 位名 | 说明       |
|-------|----|----------|
| 31:16 | -- | 未实现：读为 0 |



|      |        |       |
|------|--------|-------|
| 15:0 | ETxCNT | 计数器数值 |
|------|--------|-------|

### 1.3 ADC

#### 1.3.1 寄存器

| 地址         | 名称            | 符号       |
|------------|---------------|----------|
| 0x40012828 | ADC 输入通道选择寄存器 | ADCINSEL |
| 0x4001282C | ADC 控制寄存器     | ADCCTL   |
| 0x40012830 | ADC 调校寄存器     | ADCTRM   |
| 0x40012834 | ADC 输出数据寄存器   | ADCDATA  |
| 0x40012838 | ADC 中断标志寄存器   | ADCIF    |

#### 1.3.2 ADC 输入通道选择寄存器

| 名称  | ADCINSEL   |       |       |        |        |       |       |       |
|-----|------------|-------|-------|--------|--------|-------|-------|-------|
| 地址  | 0x40012828 |       |       |        |        |       |       |       |
| 位   | Bit31      | Bit30 | Bit29 | Bit28  | Bit27  | Bit26 | Bit25 | Bit24 |
| 位名  | -          |       |       |        |        |       |       |       |
| 位权限 | U-0        |       |       |        |        |       |       |       |
| 位   | Bit23      | Bit22 | Bit21 | Bit20  | Bit19  | Bit18 | Bit17 | Bit16 |
| 位名  | -          |       |       |        |        |       |       |       |
| 位权限 | U-0        |       |       |        |        |       |       |       |
| 位   | Bit15      | Bit14 | Bit13 | Bit12  | Bit11  | Bit10 | Bit9  | Bit8  |
| 位名  | -          |       |       |        |        |       |       |       |
| 位权限 | U-0        |       |       |        |        |       |       |       |
| 位   | Bit7       | Bit6  | Bit5  | Bit4   | Bit3   | Bit2  | Bit1  | Bit0  |
| 位名  | -          |       | BUFEN | BUFBYP | BUFSEL |       |       |       |



|     |     |       |       |          |
|-----|-----|-------|-------|----------|
| 位权限 | U-0 | R/W-0 | R/W-0 | R/W-0000 |
|-----|-----|-------|-------|----------|

| Bit  | 助记符    | 功能描述  |
|------|--------|---|
| 31:6 | --     | RFU: 未实现, 读为 0  |
| 5    | BUFEN  | ADC 输入通道 Buffer 使能  |
| 4    | BUFBYP | ADC 输入 Buffer Bypass<br>使用 ADC 测量外部信号输入时, 不要 Bypass Buffer 使用 ADC 测量电源电压时, 必须将此位置 1   |
| 3:0  | BUFSEL | ADC 输入通道选择<br>0000-0101: 保留, 禁止使用<br>0110: VDD 主电源<br>0111: 保留<br>1000: ADC_IN1 (PC12)<br>1001: ADC_IN2 (PC13)<br>1010: ADC_IN3 (PD0)<br>1011: ADC_IN4 (PD1)<br>1100: ADC_IN5 (PF6)<br>1101: ADC_IN6 (PC15)<br>1110: ADC_IN7 (PB2)<br>1111: ADC_IN8 (PB3) |

### 1.3.3 ADC 控制寄存器

| 名称  | ADCCTL     |       |       |       |       |       |       |       |
|-----|------------|-------|-------|-------|-------|-------|-------|-------|
| 地址  | 0x4001282C |       |       |       |       |       |       |       |
| 位   | Bit31      | Bit30 | Bit29 | Bit28 | Bit27 | Bit26 | Bit25 | Bit24 |
| 位名  | -          |       |       |       |       |       |       |       |
| 位权限 | U-0        |       |       |       |       |       |       |       |
| 位   | Bit23      | Bit22 | Bit21 | Bit20 | Bit19 | Bit18 | Bit17 | Bit16 |



|     |        |       |       |       |       |       |             |        |
|-----|--------|-------|-------|-------|-------|-------|-------------|--------|
| 位名  | -      |       |       |       |       |       |             |        |
| 位权限 | U-0    |       |       |       |       |       |             |        |
| 位   | Bit15  | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9        | Bit8   |
| 位名  | -      |       |       |       |       |       |             |        |
| 位权限 | U-0    |       |       |       |       |       |             |        |
| 位   | Bit7   | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1        | Bit0   |
| 位名  | ADC_IE | -     |       |       |       |       | ADC_VANA_EN | ADC_EN |
| 位权限 | R/W-0  | U-0   |       |       |       |       | R/W-0       | R/W-0  |

| Bit  | 助记符         | 功能描述  |
|------|-------------|---|
| 31:8 | --          | RFU: 未实现, 读为 0                                |
| 7    | ADC_IE      | ADC 中断使能                                      |
| 6:2  | --          | RFU: 未实现, 读为 0                                |
| 1    | ADC_VANA_EN | 外部电压通道使能<br>0: ADC 用作温度传感器<br>1: ADC 用于测量外部电压 |
| 0    | ADC_EN      | ADC 使能信号<br>0: ADC 不使能<br>1: ADC 使能           |

### 1.3.4 ADC 调校寄存器

|     |            |       |       |       |       |       |       |       |
|-----|------------|-------|-------|-------|-------|-------|-------|-------|
| 名称  | ADCTRIM    |       |       |       |       |       |       |       |
| 地址  | 0x40012830 |       |       |       |       |       |       |       |
| 位   | Bit31      | Bit30 | Bit29 | Bit28 | Bit27 | Bit26 | Bit25 | Bit24 |
| 位名  | -          |       |       |       |       |       |       |       |
| 位权限 | U-0        |       |       |       |       |       |       |       |



|     |               |       |       |       |       |                 |       |       |
|-----|---------------|-------|-------|-------|-------|-----------------|-------|-------|
| 位   | Bit23         | Bit22 | Bit21 | Bit20 | Bit19 | Bit18           | Bit17 | Bit16 |
| 位名  | -             |       |       |       |       |                 |       |       |
| 位权限 | U-0           |       |       |       |       |                 |       |       |
| 位   | Bit15         | Bit14 | Bit13 | Bit12 | Bit11 | Bit10           | Bit9  | Bit8  |
| 位名  | -             |       |       |       |       | a lc_trim[10:8] |       |       |
| 位权限 | U-0           |       |       |       |       | RW-000          |       |       |
| 位   | Bit7          | Bit6  | Bit5  | Bit4  | Bit3  | Bit2            | Bit1  | Bit0  |
| 位名  | ADC_TRIM[7:0] |       |       |       |       |                 |       |       |
| 位权限 | RW-00000000   |       |       |       |       |                 |       |       |

| Bit   | 助记符      | 功能描述           |
|-------|----------|----------------|
| 31:11 | --       | RFU: 未实现, 读为 0 |
| 10:0  | ADC_TRIM | ADC TRIM 值     |

### 1.3.5 ADC 输出数据寄存器

|     |            |       |       |       |       |                |       |       |
|-----|------------|-------|-------|-------|-------|----------------|-------|-------|
| 名称  | ADCDATA    |       |       |       |       |                |       |       |
| 地址  | 0x40012834 |       |       |       |       |                |       |       |
| 位   | Bit31      | Bit30 | Bit29 | Bit28 | Bit27 | Bit26          | Bit25 | Bit24 |
| 位名  | -          |       |       |       |       |                |       |       |
| 位权限 | U-0        |       |       |       |       |                |       |       |
| 位   | Bit23      | Bit22 | Bit21 | Bit20 | Bit19 | Bit18          | Bit17 | Bit16 |
| 位名  | -          |       |       |       |       |                |       |       |
| 位权限 | U-0        |       |       |       |       |                |       |       |
| 位   | Bit15      | Bit14 | Bit13 | Bit12 | Bit11 | Bit10          | Bit9  | Bit8  |
| 位名  | -          |       |       |       |       | ADC_DATA[10:8] |       |       |



|     |               |      |      |      |      |      |         |      |
|-----|---------------|------|------|------|------|------|---------|------|
| 位权限 | U-0           |      |      |      |      |      | R/Dy-00 |      |
| 位   | Bit7          | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1    | Bit0 |
| 位名  | ADC_DATA[7:0] |      |      |      |      |      |         |      |
| 位权限 | R/Dy-00000000 |      |      |      |      |      |         |      |

| Bit   | 助记符      | 功能描述           |
|-------|----------|----------------|
| 31:11 | --       | RFU: 未实现, 读为 0 |
| 10:0  | ADC_DATA | ADC 输出数据       |

### 1.3.6 ADC 中断标志寄存器

|     |            |       |       |       |       |       |              |         |
|-----|------------|-------|-------|-------|-------|-------|--------------|---------|
| 名称  | ADCIF      |       |       |       |       |       |              |         |
| 地址  | 0x40012838 |       |       |       |       |       |              |         |
| 位   | Bit31      | Bit30 | Bit29 | Bit28 | Bit27 | Bit26 | Bit25        | Bit24   |
| 位名  | -          |       |       |       |       |       |              |         |
| 位权限 | U-0        |       |       |       |       |       |              |         |
| 位   | Bit23      | Bit22 | Bit21 | Bit20 | Bit19 | Bit18 | Bit17        | Bit16   |
| 位名  | -          |       |       |       |       |       |              |         |
| 位权限 | U-0        |       |       |       |       |       |              |         |
| 位   | Bit15      | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9         | Bit8    |
| 位名  | -          |       |       |       |       |       |              |         |
| 位权限 | U-0        |       |       |       |       |       |              |         |
| 位   | Bit7       | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1         | Bit0    |
| 位名  | -          |       |       |       |       |       | ADC_DO<br>NE | ADC_IF  |
| 位权限 | U-0        |       |       |       |       |       | R/Dy-0       | R/W1C-0 |



| Bit  | 助记符      | 功能描述   |
|------|----------|--|
| 31:2 | --       | RFU: 未实现, 读为 0                                     |
| 1    | ADC_DONE | ADC 转换完成输出, 软件只读<br>转换完成后此信号保持为高电平, 只有关闭 ADC 才会清 0 |
| 0    | ADC_IF   | ADC 转换完成中断标志, 硬件置位, 软件写 1 清零, 写 0 无效               |

## 2 库函数

### 2.1 RCC 库函数

| 序号 | 函数名                                | 描述                |
|----|------------------------------------|-------------------|
| 1  | RCC_SysClk_Init                    | 系统时钟初始化           |
| 2  | RCC_RCHF_Init                      | RCHF 时钟初始化        |
| 3  | RCC_PLL_Init                       | PLL 锁相环初始化        |
| 4  | RCC_Deinit                         | 芯片时钟初始化为缺省值       |
| 5  | RCC_PERCLK_SetableEx               | 设置外设时钟使能          |
| 6  | RCC_Init_RCHF_Trim                 | 设置 RCHF 常温校准值载入   |
| 7  | RCC_GetClocksFreq                  | 获取芯片已设置的不同时钟频率    |
| 8  | RCC_SYSCLKSEL_SYSCLKSEL_Set        | 系统时钟源选择           |
| 9  | RCC_SYSCLKSEL_SYSCLKSEL_Get        | 获取系统时钟源选择的设置      |
| 10 | RCC_RCHFCON_FSEL_Set               | 设置 RCHF 时钟频率选择    |
| 11 | RCC_RCHFCON_FSEL_Get               | 获取 RCHF 时钟频率选择的设置 |
| 12 | RCC_RCHFCON_RCHFEN_Setable         | 设置 RCHF 使能        |
| 13 | RCC_RCHFCON_RCHFEN_Getable         | 获取 RCHF 使能状态      |
| 14 | RCC_RCHFTRIM_Write                 | 设置 RCHF 频率调校      |
| 15 | RCC_RCHFTRIM_Read                  | 获取 RCHF 频率调校设置值   |
| 16 | RCC_RCLFTRIM_Write                 | 设置 RCLF 频率调校      |
| 17 | RCC_RCLFTRIM_Read                  | 获取 RCLF 频率调校设置值   |
| 18 | RCC_SYSCLKSEL_LPM_RCLP_OFF_Setable | 休眠模式下设置 RCLP      |
| 19 | RCC_SYSCLKSEL_LPM_RCLP_OFF_Getable | 获取休眠模式下对 RCLP 的设置 |



|    |                                    |                         |
|----|------------------------------------|-------------------------|
| 20 | RCC_SYSCLOCKSEL_LSCKSEL_Set        | 无晶体模式下 LSCLK 时钟源选择      |
| 21 | RCC_SYSCLOCKSEL_LSCKSEL_Get        | 获取无晶体模式下 LSCLK 时钟源选择    |
| 22 | RCC_RCLPTRIM_Write                 | 设置 RCLP 频率调校            |
| 23 | RCC_RCLPTRIM_Read                  | 获取 RCLP 频率调校设置值         |
| 24 | RCC_RCLPCON_RCLP_EN_B_Chk          | 设置 RCLP 使能选择            |
| 25 | RCC_XTLFIPW_XTLFIPW_Set            | 设置 XTLF 工作电流选择          |
| 26 | RCC_XTLFIPW_XTLFIPW_Get            | 获取 XTLF 工作电流选择设定        |
| 27 | RCC_PLLCON_PLLDB_Set               | 设置 PLL 倍频数              |
| 28 | RCC_PLLCON_PLLDB_Get               | 获取 PLL 倍频值的设定           |
| 29 | RCC_PLLCON_PLLOSEL_Set             | 设置 PLL 输出选择             |
| 30 | RCC_PLLCON_PLLOSEL_Get             | 获取 PLL 输出选择设定           |
| 31 | RCC_PLLCON_PLLINSEL_Set            | 设置 PLL 输入选择             |
| 32 | RCC_PLLCON_PLLINSEL_Get            | 获取 PLL 输入选择设定           |
| 33 | RCC_PLLCON_PLEN_Setable            | 设置 PLL 使能设定             |
| 34 | RCC_PLLCON_PLEN_Getable            | 获取 PLL 使能状态             |
| 35 | RCC_RCLFCON_RCLFENB_Setable        | 设置 RCLF 使能设定            |
| 36 | RCC_RCLFCON_RCLFENB_Getable        | 获取 RCLF 使能状态            |
| 37 | RCC_SYSCLOCKSEL_AHBPRESET_Set      | 设置 AHB 时钟分频选择           |
| 38 | RCC_SYSCLOCKSEL_AHBPRESET_Get      | 获取 AHB 时钟分频设定           |
| 39 | RCC_MPRIL_MPRIL_Set                | 设置 AHB Master 优先级设置     |
| 40 | RCC_MPRIL_MPRIL_Get                | 获取 AHB Master 优先级设置值    |
| 41 | RCC_SYSCLOCKSEL_APBPRESET_Set      | 设置 APB 时钟分频选择           |
| 42 | RCC_SYSCLOCKSEL_APBPRESET_Get      | 获取 APB 时钟分频设定           |
| 43 | RCC_SYSCLOCKSEL_EXTICKSEL_Set      | 设置 EXTI 中断与数字滤波采样时钟选择   |
| 44 | RCC_SYSCLOCKSEL_EXTICKSEL_Get      | 获取 EXTI 中断与数字滤波采样时钟选择设定 |
| 45 | RCC_SYSCLOCKSEL_SLP_ENEXTI_Setable | 设置休眠模式下 EXTI 采样         |
| 46 | RCC_SYSCLOCKSEL_SLP_ENEXTI_Getable | 获取休眠模式下 EXTI 采样设置值      |
| 47 | RCC_PERCLKCON2_ADCCCKSEL_Set       | 设置 ADC 工作时钟选择           |
| 48 | RCC_PERCLKCON2_ADCCCKSEL_Get       | 获取 ADC 工作时钟选择值          |

**RCC\_SysClk\_Init**

| 函数名    | RCC_SysClk_Init   |
|--------|---|
| 函数原型   | void RCC_SysClk_Init(RCC_SYSCLK_InitTypeDef* para)  |
| 功能描述   | 系统时钟初始化   |
| 输入参数 1 | <p>系统时钟设置的相关参数：</p> <div> <p>系统时钟源选择：</p> <p>00: RCHF</p> <p>01: RCLP</p> <p>10: LSCLK</p> <p>11: PLL</p> </div> <div> <p>AHB 时钟分频选择：</p> <p>0XX: 不分频</p> <p>100: 2 分频</p> <p>101: 4 分频</p> <p>110: 8 分频</p> <p>111: 16 分频</p> </div> <div> <p>APB 时钟分频选择：</p> <p>0XX: 不分频</p> <p>100: 2 分频</p> <p>101: 4 分频</p> <p>110: 8 分频</p> <p>111: 16 分频</p> </div> <div> <p>EXTI 中断采样、IO 数字滤波时钟选择：</p> <p>0: 外部引脚中断使用 AHBCLK 采样</p> <p>1: 外部引脚中断使用 LSCLK 采样</p> </div> <div> <p>睡眠模式下 EXTI 采样设置：</p> <p>0: 睡眠模式下禁止外部引脚中断采样</p> <p>1: 睡眠模式下使能外部引脚中断采样</p> </div> <div> <p>睡眠模式下 PCLP 设置：</p> <p>0: 睡眠模式下保持 RCLP 开启</p> <p>1: 睡眠模式下禁止 RCLP</p> </div> |

**RCC\_RCHF\_Init**

| 函数名    | RCC_RCHF_Init  |
|--------|--|
| 函数原型   | void RCC_RCHF_Init(RCC_RCHF_InitTypeDef* para)   |
| 功能描述   | RCHF 时钟初始化   |
| 输入参数 1 | <div>RCHF 时钟设置的相关参数：</div> <div><div>RCHF 时钟频率选择：<br/>00: 8MHz<br/>01: 16MHz<br/>10: 24MHz<br/>11: RFU</div><div>RCHF 使能：<br/>DISABLE: 禁止 RCHF<br/>ENABLE: 使能 RCHF</div></div> |
| 输出参数   | 无  |
| 返回值    | 无  |

**RCC\_PLL\_Init**

| 函数名    | RCC_PLL_Init  |
|--------|---|
| 函数原型   | void RCC_PLL_Init(RCC_PLL_InitTypeDef* para)  |
| 功能描述   | PLL 初始化   |
| 输入参数 1 | <div>PLL 设置的相关参数：</div> <div>PLL 倍频比：0~1023</div> <div>输出选择：<br/>0：选择 PLL 一倍输出作为数字电路内的 PLL 时钟<br/>1：选择 PLL 二倍输出作为数字电路内的 PLL 时钟</div> <div>PLL 输入源选择：<br/>00/11：XTLF<br/>01：RCLP<br/>10：RCHF/256</div> <div>PLL 使能控制：<br/>DISABLE：禁止 PLL<br/>ENABLE：使能 PLL</div> |
| 输出参数   | 无   |



|     |   |
|-----|---|
| 返回值 | 无 |
|-----|---|

**RCC\_Deinit**

| 函数名    | RCC_Deinit            |
|--------|-----------------------|
| 函数原型   | void RCC_Deinit(void) |
| 功能描述   | 芯片时钟初始化为缺省值           |
| 输入参数 1 | 无                     |
| 输出参数   | 无                     |
| 返回值    | 无                     |

**RCC\_PERCLK\_SetableEx**

| 函数名    | RCC_PERCLK_SetableEx  |
|--------|---|
| 函数原型   | void RCC_PERCLK_SetableEx(uint32_t periph_def, FunState NewState) |
| 功能描述   | 设置外设时钟使能  |
| 输入参数 1 | 外设模块标号: 0x01XXXXXX~0x04XXXXXX                                     |
| 输入参数 2 | 外设时钟使能状态:<br>DISABLE: 禁止外设时钟<br>ENABLE: 使能外设时钟                    |
| 输出参数   | 无   |
| 返回值    | 无   |

**RCC\_Init\_RCHF\_Trim**

| 函数名    | RCC_Init_RCHF_Trim                               |
|--------|--|
| 函数原型   | void RCC_Init_RCHF_Trim( uint8_t ClkMode )       |
| 功能描述   | 设置 RCHF 常温校准值载入                                  |
| 输入参数 1 | RCHF 频率设置:<br>0X01: 8M<br>0X02: 16M<br>0X03: 24M |



|      |                     |
|------|---------------------|
|      | 0X04: 32M<br>其他: 8M |
| 输出参数 | 无                   |
| 返回值  | 无                   |

**RCC\_GetClocksFreq**

| 函数名                        | RCC_GetClocksFreq  |                          |                            |                            |                           |                         |
|----------------------------|--|--------------------------|----------------------------|----------------------------|---------------------------|-------------------------|
| 函数原型                       | void RCC_GetClocksFreq(RCC_ClocksType* para)   |                          |                            |                            |                           |                         |
| 功能描述                       | 获取芯片已设置的不同时钟频率   |                          |                            |                            |                           |                         |
| 输入参数 1                     | 无  |                          |                            |                            |                           |                         |
| 输出参数                       | <div>不同时钟的频率参数：<table><tr><td>SYSCLK_Frequency: 系统时钟频率</td></tr><tr><td>AHBCLK_Frequency: AHB 时钟频率</td></tr><tr><td>APBCLK_Frequency: APB 时钟频率</td></tr><tr><td>RCHF_Frequency: RCHF 时钟频率</td></tr><tr><td>PLL_Frequency: PLL 时钟频率</td></tr></table></div> | SYSCLK_Frequency: 系统时钟频率 | AHBCLK_Frequency: AHB 时钟频率 | APBCLK_Frequency: APB 时钟频率 | RCHF_Frequency: RCHF 时钟频率 | PLL_Frequency: PLL 时钟频率 |
| SYSCLK_Frequency: 系统时钟频率   |  |                          |                            |                            |                           |                         |
| AHBCLK_Frequency: AHB 时钟频率 |  |                          |                            |                            |                           |                         |
| APBCLK_Frequency: APB 时钟频率 |  |                          |                            |                            |                           |                         |
| RCHF_Frequency: RCHF 时钟频率  |  |                          |                            |                            |                           |                         |
| PLL_Frequency: PLL 时钟频率    |  |                          |                            |                            |                           |                         |
| 返回值                        | 无  |                          |                            |                            |                           |                         |

**RCC\_SYSCLKSEL\_SYSCLKSEL\_Set**

| 函数名    | RCC_SYSCLKSEL_SYSCLKSEL_Set                                 |
|--------|---|
| 函数原型   | void RCC_SYSCLKSEL_SYSCLKSEL_Set(uint32_t SetValue)         |
| 功能描述   | 设置系统时钟源选择   |
| 输入参数 1 | 系统时钟源选择的设置:<br>00: RCHF<br>01: RCLP<br>10: LSCLK<br>11: PLL |
| 输出参数   | 无   |
| 返回值    | 无   |

**RCC\_SYCLKSEL\_SYCLKSEL\_Get**

| 函数名    | RCC_SYCLKSEL_SYCLKSEL_Get                                   |
|--------|---|
| 函数原型   | uint32_t RCC_SYCLKSEL_SYCLKSEL_Get(void)                    |
| 功能描述   | 获取系统时钟源选择的设置  |
| 输入参数 1 | 无   |
| 输出参数   | 无   |
| 返回值    | 系统时钟源选择的设置：<br>00: RCHF<br>01: RCLP<br>10: LSCLK<br>11: PLL |

**RCC\_RCHFCON\_FSEL\_Set**

| 函数名    | RCC_RCHFCON_FSEL_Set  |
|--------|---|
| 函数原型   | void RCC_RCHFCON_FSEL_Set(uint32_t SetValue)                    |
| 功能描述   | 设置 RCHF 时钟频率选择  |
| 输入参数 1 | RCHF 时钟频率选择设置：<br>00: 8MHz<br>01: 16MHz<br>10: 24MHz<br>11: RFU |
| 输出参数   | 无   |
| 返回值    | 无   |

**RCC\_RCHFCON\_FSEL\_Get**

| 函数名    | RCC_RCHFCON_FSEL_Get                |
|--------|-------------------------------------|
| 函数原型   | uint32_t RCC_RCHFCON_FSEL_Get(void) |
| 功能描述   | 获取 RCHF 时钟频率选择的设置                   |
| 输入参数 1 | 无                                   |
| 输出参数   | 无                                   |



|     |   |
|-----|---|
| 返回值 | RCHF 时钟频率选择设置:<br>00: 8MHz<br>01: 16MHz<br>10: 24MHz<br>11: RFU |
|-----|---|

**RCC\_RCHFCON\_RCHFEN\_Setable**

| 函数名    | RCC_RCHFCON_RCHFEN_Setable                              |
|--------|---|
| 函数原型   | void RCC_RCHFCON_RCHFEN_Setable(FunState NewState)      |
| 功能描述   | 设置 RCHF 使能  |
| 输入参数 1 | RCHF 使能状态:<br>DISABLE: 禁止 RCHF 时钟<br>ENABLE: 使能 RCHF 时钟 |
| 输出参数   | 无   |
| 返回值    | 无   |

**RCC\_RCHFCON\_RCHFEN\_Getable**

| 函数名    | RCC_RCHFCON_RCHFEN_Getable                              |
|--------|---|
| 函数原型   | FunState RCC_RCHFCON_RCHFEN_Getable(void)               |
| 功能描述   | 获取 RCHF 使能状态  |
| 输入参数 1 | 无   |
| 输出参数   | 无   |
| 返回值    | RCHF 使能状态:<br>DISABLE: 禁止 RCHF 时钟<br>ENABLE: 使能 RCHF 时钟 |

**RCC\_RCHFTRIM\_Write**

| 函数名  | RCC_RCHFTRIM_Write                         |
|------|--|
| 函数原型 | void RCC_RCHFTRIM_Write(uint32_t SetValue) |
| 功能描述 | 设置 RCHF 频率调校                               |



|        |                           |
|--------|---------------------------|
| 输入参数 1 | RCHF 频率调校的设置数值: 0X00~0X3F |
| 输出参数   | 无                         |
| 返回值    | 无                         |

**RCC\_RCHFTRIM\_Read**

| 函数名    | RCC_RCHFTRIM_Read                |
|--------|----------------------------------|
| 函数原型   | uint32_t RCC_RCHFTRIM_Read(void) |
| 功能描述   | 获取 RCHF 频率调校设置值                  |
| 输入参数 1 | 无                                |
| 输出参数   | 无                                |
| 返回值    | RCHF 频率调校的设置数值: 0X00~0X3F        |

**RCC\_RCLFTRIM\_Write**

| 函数名    | RCC_RCLFTRIM_Write                         |
|--------|--|
| 函数原型   | void RCC_RCLFTRIM_Write(uint32_t SetValue) |
| 功能描述   | 设置 RCLF 频率调校                               |
| 输入参数 1 | RCLF 频率调校值                                 |
| 输出参数   | 无  |
| 返回值    | 无  |

**RCC\_RCLFTRIM\_Read**

| 函数名    | RCC_RCLFTRIM_Read                |
|--------|----------------------------------|
| 函数原型   | uint32_t RCC_RCLFTRIM_Read(void) |
| 功能描述   | 获取 RCLF 频率调校设置值                  |
| 输入参数 1 | 无                                |
| 输出参数   | 无                                |
| 返回值    | RCLF 频率调校值                       |

**RCC\_SYSCLKSEL\_LPM\_RCLP\_OFF\_Setable**

| 函数名 | RCC_SYSCLKSEL_LPM_RCLP_OFF_Setable |
|-----|------------------------------------|
|-----|------------------------------------|



|        |   |
|--------|---|
| 函数原型   | void RCC_SYCLKSEL_LPM_RCLP_OFF_Setable(FunState NewState)     |
| 功能描述   | 设置休眠模式下对 RCLP 操作  |
| 输入参数 1 | 休眠模式下 RCLP 使能状态:<br>DISABLE: 禁止 RCLP 时钟<br>ENABLE: 使能 RCLP 时钟 |
| 输出参数   | 无   |
| 返回值    | 无   |

### RCC\_SYCLKSEL\_LPM\_RCLP\_OFF\_Getable

| 函数名    | RCC_SYCLKSEL_LPM_RCLP_OFF_Getable                             |
|--------|---|
| 函数原型   | FunState RCC_SYCLKSEL_LPM_RCLP_OFF_Getable(void)              |
| 功能描述   | 获取休眠模式下对 RCLP 的设置   |
| 输入参数 1 | 无   |
| 输出参数   | 无   |
| 返回值    | 休眠模式下 RCLP 使能状态:<br>DISABLE: 禁止 RCLP 时钟<br>ENABLE: 使能 RCLP 时钟 |

### RCC\_SYCLKSEL\_LSCKSEL\_Set

| 函数名    | RCC_SYCLKSEL_LSCKSEL_Set  |
|--------|---|
| 函数原型   | void RCC_SYCLKSEL_LSCKSEL_Set(uint32_t SetValue)  |
| 功能描述   | 设置无晶体模式下 LSCLK 时钟源选择  |
| 输入参数 1 | 无晶体模式下 LSCLK 时钟源选择:<br>RCC_SYCLKSEL_LSCKSEL_RCLP: RCLP<br>RCC_SYCLKSEL_LSCKSEL_RCLF: RCLF |
| 输出参数   | 无   |
| 返回值    | 无   |

**RCC\_SYSCCLKSEL\_LSCKSEL\_Get**

| 函数名    | RCC_SYSCCLKSEL_LSCKSEL_Get  |
|--------|---|
| 函数原型   | uint32_t RCC_SYSCCLKSEL_LSCKSEL_Get(void)   |
| 功能描述   | 获取无晶体模式下 LSCLK 时钟源选择  |
| 输入参数 1 | 无   |
| 输出参数   | 无   |
| 返回值    | 无晶体模式下 LSCLK 时钟源选择：<br>RCC_SYSCCLKSEL_LSCKSEL_RCLP: RCLP<br>RCC_SYSCCLKSEL_LSCKSEL_RCLF: RCLF |

**RCC\_RCLPTRIM\_Write**

| 函数名    | RCC_RCLPTRIM_Write                         |
|--------|--|
| 函数原型   | void RCC_RCLPTRIM_Write(uint32_t SetValue) |
| 功能描述   | 设置 RCLP 频率调校                               |
| 输入参数 1 | RCLP 频率调校值: 0-F                            |
| 输出参数   | 无  |
| 返回值    | 无  |

**RCC\_RCLPTRIM\_Read**

| 函数名    | RCC_RCLPTRIM_Read                |
|--------|----------------------------------|
| 函数原型   | uint32_t RCC_RCLPTRIM_Read(void) |
| 功能描述   | 获取 RCLP 频率调校设置值                  |
| 输入参数 1 | 无                                |
| 输出参数   | 无                                |
| 返回值    | RCLP 频率调校值: 0-F                  |

**RCC\_RCLPCON\_RCLP\_EN\_B\_Chk**

| 函数名  | RCC_RCLPCON_RCLP_EN_B_Chk                  |
|------|--|
| 函数原型 | FlagStatus RCC_RCLPCON_RCLP_EN_B_Chk(void) |
| 功能描述 | 获取 RCLP 使能状态                               |



|        |   |
|--------|---|
| 输入参数 1 | 无   |
| 输出参数   | 无   |
| 返回值    | RCLP 使能状态:<br>DISABLE: 关闭 RCLP<br>ENABLE: 使能 RCLP |

**RCC\_XTLFIPW\_XTLFIPW\_Set**

| 函数名    | RCC_XTLFIPW_XTLFIPW_Set  |
|--------|--|
| 函数原型   | void RCC_XTLFIPW_XTLFIPW_Set(uint32_t SetValue)  |
| 功能描述   | 设置 XTLF 工作电流选择   |
| 输入参数 1 | XTLF 工作电流:<br>000: 450nA<br>001: 400nA<br>010: 350nA<br>011: 300nA<br>100: 250nA<br>101: 200nA<br>110: 150nA<br>111: 100nA |
| 输出参数   | 无  |
| 返回值    | 无  |

**RCC\_XTLFIPW\_XTLFIPW\_Get**

| 函数名    | RCC_XTLFIPW_XTLFIPW_Get                |
|--------|--|
| 函数原型   | uint32_t RCC_XTLFIPW_XTLFIPW_Get(void) |
| 功能描述   | 获取 XTLF 工作电流选择设定                       |
| 输入参数 1 | 无                                      |
| 输出参数   | 无                                      |
| 返回值    | XTLF 工作电流:<br>000: 450nA               |



|  |            |
|--|------------|
|  | 001: 400nA |
|  | 010: 350nA |
|  | 011: 300nA |
|  | 100: 250nA |
|  | 101: 200nA |
|  | 110: 150nA |
|  | 111: 100nA |

**RCC\_PLLCON\_PLLDB\_Set**

| 函数名    | RCC_PLLCON_PLLDB_Set                         |
|--------|--|
| 函数原型   | void RCC_PLLCON_PLLDB_Set(uint32_t SetValue) |
| 功能描述   | 设置 PLL 倍频数设置                                 |
| 输入参数 1 | PLL 倍频数: 0-3FFFF                             |
| 输出参数   | 无  |
| 返回值    | 无  |

**RCC\_PLLCON\_PLLDB\_Get**

| 函数名    | RCC_PLLCON_PLLDB_Get                |
|--------|-------------------------------------|
| 函数原型   | uint32_t RCC_PLLCON_PLLDB_Get(void) |
| 功能描述   | 获取 PLL 倍频值的设定                       |
| 输入参数 1 | 无                                   |
| 输出参数   | 无                                   |
| 返回值    | PLL 倍频数: 0-3FFFF                    |

**RCC\_PLLCON\_PLLOSEL\_Set**

| 函数名    | RCC_PLLCON_PLLOSEL_Set                         |
|--------|--|
| 函数原型   | void RCC_PLLCON_PLLOSEL_Set(uint32_t SetValue) |
| 功能描述   | 设置 PLL 输出选择                                    |
| 输入参数 1 | PLL 输出选择:<br>0: 选择 PLL 一倍输出作为数字电路内的 PLL 时钟     |



|      |                               |
|------|-------------------------------|
|      | 1: 选择 PLL 两倍输出作为数字电路内的 PLL 时钟 |
| 输出参数 | 无                             |
| 返回值  | 无                             |

**RCC\_PLLCON\_PLLOSEL\_Get**

| 函数名    | RCC_PLLCON_PLLOSEL_Get  |
|--------|---|
| 函数原型   | uint32_t RCC_PLLCON_PLLOSEL_Get(void)                                       |
| 功能描述   | 获取 PLL 输出选择设定   |
| 输入参数 1 | 无   |
| 输出参数   | 无   |
| 返回值    | PLL 输出选择:<br>0: 选择 PLL 一倍输出作为数字电路内的 PLL 时钟<br>1: 选择 PLL 两倍输出作为数字电路内的 PLL 时钟 |

**RCC\_PLLCON\_PLLINSEL\_Set**

| 函数名    | RCC_PLLCON_PLLINSEL_Set                               |
|--------|---|
| 函数原型   | void RCC_PLLCON_PLLINSEL_Set(uint32_t SetValue)       |
| 功能描述   | 设置 PLL 输入选择   |
| 输入参数 1 | PLL 输入源选择:<br>00/11: XTLF<br>01: RCLP<br>10: RCHF/256 |
| 输出参数   | 无   |
| 返回值    | 无   |

**RCC\_PLLCON\_PLLINSEL\_Get**

| 函数名    | RCC_PLLCON_PLLINSEL_Get                |
|--------|--|
| 函数原型   | uint32_t RCC_PLLCON_PLLINSEL_Get(void) |
| 功能描述   | 获取 PLL 输入选择设定                          |
| 输入参数 1 | 无                                      |



|      |   |
|------|---|
| 输出参数 | 无   |
| 返回值  | PLL 输入源选择:<br>00/11: XTLE<br>01: RCLP<br>10: RCHF/256 |

**RCC\_PLLCON\_PPLEN\_Setable**

| 函数名    | RCC_PLLCON_PPLEN_Setable                         |
|--------|--|
| 函数原型   | void RCC_PLLCON_PPLEN_Setable(FunState NewState) |
| 功能描述   | 设置 PLL 使能  |
| 输入参数 1 | PLL 使能状态<br>DISABLE: 禁止 PLL<br>ENABLE: 使能 PLL    |
| 输出参数   | 无  |
| 返回值    | 无  |

**RCC\_PLLCON\_PPLEN\_Getable**

| 函数名    | RCC_PLLCON_PPLEN_Getable                      |
|--------|---|
| 函数原型   | FunState RCC_PLLCON_PPLEN_Getable(void)       |
| 功能描述   | 获取 PLL 使能状态                                   |
| 输入参数 1 | 无   |
| 输出参数   | 无   |
| 返回值    | PLL 使能状态<br>DISABLE: 禁止 PLL<br>ENABLE: 使能 PLL |

**RCC\_RCLFCON\_RCLFENB\_Setable**

| 函数名  | RCC_RCLFCON_RCLFENB_Setable                         |
|------|---|
| 函数原型 | void RCC_RCLFCON_RCLFENB_Setable(FunState NewState) |
| 功能描述 | 设置 RCLF 使能  |



|        |   |
|--------|---|
| 输入参数 1 | RCLF 使能:<br>DISABLE: 禁止 RCLF<br>ENABLE: 使能 RCLF |
| 输出参数   | 无   |
| 返回值    | 无   |

**RCC\_RCLFCON\_RCLFENB\_Getable**

| 函数名    | RCC_RCLFCON_RCLFENB_Getable                     |
|--------|---|
| 函数原型   | FunState RCC_RCLFCON_RCLFENB_Getable(void)      |
| 功能描述   | 获取 RCLF 使能状态                                    |
| 输入参数 1 | 无   |
| 输出参数   | 无   |
| 返回值    | RCLF 使能:<br>DISABLE: 禁止 RCLF<br>ENABLE: 使能 RCLF |

**RCC\_SYCLKSEL\_AHBPRES\_Set**

| 函数名    | RCC_SYCLKSEL_AHBPRES_Set   |
|--------|--|
| 函数原型   | void RCC_SYCLKSEL_AHBPRES_Set(uint32_t SetValue)                             |
| 功能描述   | 设置 AHB 时钟分频选择  |
| 输入参数 1 | AHB 时钟分频选择:<br>0XX: 不分频<br>100: 2 分频<br>101: 4 分频<br>110: 8 分频<br>111: 16 分频 |
| 输出参数   | 无  |
| 返回值    | 无  |

**RCC\_SYCLKSEL\_AHBPRES\_Get**

| 函数名 | RCC_SYCLKSEL_AHBPRES_Get |
|-----|--------------------------|
|-----|--------------------------|



|        |  |
|--------|--|
| 函数原型   | uint32_t RCC_SYCLKSEL_AHBPRES_Get(void)                                      |
| 功能描述   | 获取 AHB 时钟分频设定  |
| 输入参数 1 | 无  |
| 输出参数   | 无  |
| 返回值    | AHB 时钟分频选择：<br>0XX: 不分频<br>100: 2 分频<br>101: 4 分频<br>110: 8 分频<br>111: 16 分频 |

**RCC\_MPRIL\_MPRIL\_Set**

| 函数名    | RCC_MPRIL_MPRIL_Set                           |
|--------|---|
| 函数原型   | void RCC_MPRIL_MPRIL_Set(uint32_t SetValue)   |
| 功能描述   | 设置 AHB Master 优先级设置                           |
| 输入参数 1 | AHB Master 优先级设置选择：<br>0: DMA 选择<br>1: CPU 选择 |
| 输出参数   | 无   |
| 返回值    | 无   |

**RCC\_MPRIL\_MPRIL\_Get**

| 函数名    | RCC_MPRIL_MPRIL_Get                           |
|--------|---|
| 函数原型   | uint32_t RCC_MPRIL_MPRIL_Get(void)            |
| 功能描述   | 获取 AHB Master 优先级设置值                          |
| 输入参数 1 | 无   |
| 输出参数   | 无   |
| 返回值    | AHB Master 优先级设置选择：<br>0: DMA 选择<br>1: CPU 选择 |

**RCC\_SYCLKSEL\_APBPRES\_Set**

| 函数名    | RCC_SYCLKSEL_APBPRES_Set   |
|--------|--|
| 函数原型   | void RCC_SYCLKSEL_APBPRES_Set(uint32_t SetValue)                             |
| 功能描述   | 设置 APB 时钟分频选择  |
| 输入参数 1 | APB 时钟分频选择:<br>0XX: 不分频<br>100: 2 分频<br>101: 4 分频<br>110: 8 分频<br>111: 16 分频 |
| 输出参数   | 无  |
| 返回值    | 无  |

**RCC\_SYCLKSEL\_APBPRES\_Get**

| 函数名    | RCC_SYCLKSEL_APBPRES_Get   |
|--------|--|
| 函数原型   | uint32_t RCC_SYCLKSEL_APBPRES_Get(void)                                      |
| 功能描述   | 获取 APB 时钟分频设定  |
| 输入参数 1 | 无  |
| 输出参数   | 无  |
| 返回值    | APB 时钟分频选择:<br>0XX: 不分频<br>100: 2 分频<br>101: 4 分频<br>110: 8 分频<br>111: 16 分频 |

**RCC\_SYCLKSEL\_EXTICKSEL\_Set**

| 函数名  | RCC_SYCLKSEL_EXTICKSEL_Set                         |
|------|--|
| 函数原型 | void RCC_SYCLKSEL_EXTICKSEL_Set(uint32_t SetValue) |



|        |   |
|--------|---|
| 功能描述   | 设置 EXTI 中断与数字滤波采样时钟选择   |
| 输入参数 1 | EXTI 中断采样、IO 数字滤波时钟选择：<br>0：外部引脚中断使用 AHBCLK 采样<br>1：外部引脚中断使用 LSCLK 采样 |
| 输出参数   | 无   |
| 返回值    | 无   |

### RCC\_SYSCCLKSEL\_EXTICKSEL\_Get

| 函数名    | RCC_SYSCCLKSEL_EXTICKSEL_Get  |
|--------|---|
| 函数原型   | uint32_t RCC_SYSCCLKSEL_EXTICKSEL_Get(void)                           |
| 功能描述   | 获取 EXTI 中断与数字滤波采样时钟选择设定   |
| 输入参数 1 | 无   |
| 输出参数   | 无   |
| 返回值    | EXTI 中断采样、IO 数字滤波时钟选择：<br>0：外部引脚中断使用 AHBCLK 采样<br>1：外部引脚中断使用 LSCLK 采样 |

### RCC\_SYSCCLKSEL\_SLP\_ENEXTI\_Setable

| 函数名    | RCC_SYSCCLKSEL_SLP_ENEXTI_Setable                          |
|--------|--|
| 函数原型   | void RCC_SYSCCLKSEL_SLP_ENEXTI_Setable(FunState NewState)  |
| 功能描述   | 设置休眠模式下 EXTI 采样  |
| 输入参数 1 | 睡眠模式下 EXTI 采样设置：<br>0：休眠模式下禁止外部引脚中断采样<br>1：休眠模式下使能外部引脚中断采样 |
| 输出参数   | 无  |
| 返回值    | 无  |

### RCC\_SYSCCLKSEL\_SLP\_ENEXTI\_Getable

| 函数名  | RCC_SYSCCLKSEL_SLP_ENEXTI_Getable                |
|------|--|
| 函数原型 | FunState RCC_SYSCCLKSEL_SLP_ENEXTI_Getable(void) |



|        |  |
|--------|--|
| 功能描述   | 获取休眠模式下 EXTI 采样设置  |
| 输入参数 1 | 无  |
| 输出参数   | 无  |
| 返回值    | 睡眠模式下 EXTI 采样设置：<br>0：休眠模式下禁止外部引脚中断采样<br>1：休眠模式下使能外部引脚中断采样 |

**RCC\_PERCLKCON2\_ADCCCKSEL\_Set**

| 函数名    | RCC_PERCLKCON2_ADCCCKSEL_Set   |
|--------|--|
| 函数原型   | void RCC_PERCLKCON2_ADCCCKSEL_Set(uint32_t SetValue)   |
| 功能描述   | 设置 ADC 工作时钟选择  |
| 输入参数 1 | ADC 工作时钟选择：<br>000: RCHF<br>001: RCHF/2<br>010: RCHF/4<br>011: RCHF/8<br>100: RCHF/16<br>101: RCHF/32<br>110: RCHF/64<br>111: RCLP |
| 输出参数   | 无  |
| 返回值    | 无  |

**RCC\_PERCLKCON2\_ADCCCKSEL\_Get**

| 函数名    | RCC_PERCLKCON2_ADCCCKSEL_Get                |
|--------|---|
| 函数原型   | uint32_t RCC_PERCLKCON2_ADCCCKSEL_Get(void) |
| 功能描述   | 获取 ADC 工作时钟选择值                              |
| 输入参数 1 | 无   |
| 输出参数   | 无   |
| 返回值    | ADC 工作时钟选择：                                 |



|  |              |
|--|--------------|
|  | 000: RCHF    |
|  | 001: RCHF/2  |
|  | 010: RCHF/4  |
|  | 011: RCHF/8  |
|  | 100: RCHF/16 |
|  | 101: RCHF/32 |
|  | 110: RCHF/64 |
|  | 111: RCLP    |

## 2.2 ETIM 库函数

| 序号 | 函数名                     | 描述                      |
|----|-------------------------|-------------------------|
| 1  | ETIMx_Init              | ET 初始化设置                |
| 2  | ETIMx_Deinit            | ET 初始化为缺省值              |
| 3  | ETIMx_ETxCR_CEN_Setable | 设置 ET 计数器启动             |
| 4  | ETIMx_ETxCR_CEN_Getable | 获取 ET 计数器启动状态           |
| 5  | ETIMx_ETxCR_MOD_Set     | 设置 ET 计数器工作模式           |
| 6  | ETIMx_ETxCR_MOD_Get     | 获取 ET 计数器工作模式           |
| 7  | ETIMx_ETxCR_EDGESEL_Set | 设置 ET 计数器计数模式采沿方式       |
| 8  | ETIMx_ETxCR_EDGESEL_Get | 获取 ET 计数器计数模式采沿方式       |
| 9  | ETIMx_ETxPESCALE1_Write | 设置 ET Signal1(计数源)的预分频  |
| 10 | ETIMx_ETxPESCALE1_Read  | 获取 ET Signal1(计数源) 的预分频 |
| 11 | ETIMx_ETxPESCALE2_Write | 设置 ET Signal2(计数源)的预分频  |
| 12 | ETIMx_ETxPESCALE2_Read  | 获取 ET Signal2(计数源) 的预分频 |
| 13 | ETIMx_ETxIVR_Write      | 设置 ET 计数初值              |



|    |                             |                      |
|----|-----------------------------|----------------------|
| 14 | ETIMx_ETxIVR_Read           | 获取 ET 计数初值           |
| 15 | ETIMx_ETxCMP_Write          | 设置 ET 比较值            |
| 16 | ETIMx_ETxCMP_Read           | 获取 ET 比较值            |
| 17 | ETIMx_ETxCR_PWM_Setable     | ET 计数器 PWM 输出使能设置    |
| 18 | ETIMx_ETxCR_PWM_Getable     | 获取 ET 计数器 PWM 输出使能设置 |
| 19 | ETIMx_ETxINSEL_SIG1SEL_Set  | 设置 ET 内部信号 1 源选择     |
| 20 | ETIMx_ETxINSEL_SIG1SEL_Get  | 获取 ET 内部信号 1 源选择     |
| 21 | ETIMx_ETxINSEL_SIG2SEL_Set  | 设置 ET 内部信号 2 源选择     |
| 22 | ETIMx_ETxINSEL_SIG2SEL_Get  | 获取 ET 内部信号 2 源选择     |
| 23 | ETIMx_ETxCR_CAPEDGE_Set     | 设置 ET 计数器捕捉沿选择       |
| 24 | ETIMx_ETxCR_CAPEDGE_Get     | 获取 ET 计数器捕捉沿选择设置     |
| 25 | ETIMx_ETxCR_CAPMOD_Set      | 设置 ET 计数器捕捉模式        |
| 26 | ETIMx_ETxCR_CAPMOD_Get      | 获取 ET 计数器捕捉模式        |
| 27 | ETIMx_ETxCR_CAPONCE_Setable | 设置 ET 计数器单次捕捉有效      |
| 28 | ETIMx_ETxCR_CAPONCE_Getable | 获取 ET 计数器单次捕捉有效设置    |
| 29 | ETIMx_ETxCR_CAPCLR_Setable  | 设置 ET 计数器带清零捕捉模式     |
| 30 | ETIMx_ETxCR_CAPCLR_Getable  | 获取 ET 计数器带清零捕捉模式设置   |
| 31 | ETIMx_ETxINSEL_GRP1SEL_Set  | 设置 ET GROUP1 信号选择    |
| 32 | ETIMx_ETxINSEL_GRP1SEL_Get  | 获取 ET GROUP1 信号选择    |
| 33 | ETIMx_ETxINSEL_GRP2SEL_Set  | 设置 ET GROUP2 信号选择    |
| 34 | ETIMx_ETxINSEL_GRP2SEL_Get  | 获取 ET GROUP2 信号选择    |
| 35 | ETIMx_ETxCR_CASEN_Setable   | 设置 ET 计数器级联功能        |
| 36 | ETIMx_ETxCR_CASEN_Getable   | 获取 ET 计数器级联功能设置      |
| 37 | ETIMx_ETxIE_CMPIE_Setable   | 设置 ET 比较中断           |
| 38 | ETIMx_ETxIE_CMPIE_Getable   | 获取 ET 比较中断设置         |
| 39 | ETIMx_ETxIE_CAPIE_Setable   | 设置 ET 捕捉中断           |



|    |                           |                         |
|----|---------------------------|-------------------------|
| 40 | ETIMx_ETxIE_CAPIE_Getable | 获取 ET 捕捉中断              |
| 41 | ETIMx_ETxIE_OVIE_Setable  | 设置 ET 溢出中断              |
| 42 | ETIMx_ETxIE_OVIE_Getable  | 获取 ET 溢出中断              |
| 43 | ETIMx_ETxIF_CMPIF_Clr     | 清除 ET 比较中断标志            |
| 44 | ETIMx_ETxIF_CMPIF_Chk     | 获取 ET 比较中断标志            |
| 45 | ETIMx_ETxIF_CAPIF_Clr     | 清除 ET 产生捕捉信号标志          |
| 46 | ETIMx_ETxIF_CAPIF_Chk     | 获取 ET 产生捕捉信号标志          |
| 47 | ETIMx_ETxIF_OVIF_Clr      | 清除 ET 产生溢出信号标志          |
| 48 | ETIMx_ETxIF_OVIF_Chk      | 获取 ET 产生溢出信号标志          |
| 49 | ETIMx_ETxIF_EDGESTA_Chk   | 获取 ET 捕捉沿状态             |
| 50 | ETIMx_ETxCR_EXFLT_Setable | ET 计数器外部输入引脚数字滤波功能设置    |
| 51 | ETIMx_ETxCR_EXFLT_Getable | 获取 ET 计数器外部输入引脚数字滤波功能设置 |
| 52 | ETIMx_ETxCNT_Read         | 获取 ET 计数器数值             |

### ETIMx\_Init

|        |  |
|--------|--|
| 函数名    | ETIMx_Init   |
| 函数原型   | void ETIMx_Init(ETIMx_Type* ETIMx, ETIM_InitTypeDef* para)   |
| 功能描述   | ET 初始化配置   |
| 输入参数 1 | 定时器名称: ETIM1/ETIM2/ETIM3/ETIM4   |
| 输入参数 2 | <div>ETIM 配置参数:</div> <div><div>信号来源参数:</div><div><div>内部信号 1 源选择:</div><div>0: 扩展定时器 3 的内部信号 1 选择 GROUP2</div><div>1: 扩展定时器 3 的内部信号 1 选择 GROUP1</div></div><div><div>内部信号 2 源选择:</div><div>0: 扩展定时器 3 的内部信号 2 选择 GROUP2</div><div>1: 扩展定时器 3 的内部信号 2 选择 GROUP1</div></div><div><div>GROUP1 信号选择控制:</div><div>ET1</div><div>00: APBCLK</div></div></div> |



|  |  |
|--|--|
|  | 01: XTLF<br>10: RCLP<br>11: ET1_IN0<br>ET2<br>00: APBCLK<br>01: XTLF<br>10: RCLP<br>11: ET2_IN0<br>ET3<br>00: APBCLK<br>01: ET3_IN0<br>10: RTCSEC<br>11: RCLP<br>ET4<br>00: APBCLK<br>01: ET4_IN0<br>10: RTC64HZ<br>11: LPTO   |
|  | GROUP2 信号选择控制:<br>ET1<br>000: UART0_RX<br>001: UART1_RX<br>010: XTLF<br>011: ET1_IN1<br>100: ET1_IN2<br>101: CMP1O<br>110: CMP2O<br>111: LPTO<br>ET2<br>000: UART2_RX<br>001: UART3_RX<br>010: XTLF<br>011: ET2_IN1<br>100: ET2_IN2<br>101: CMP1O<br>110: CMP2O<br>111: LPTO |



|  |                               |
|--|-------------------------------|
|  | ET3                           |
|  | 000: ET3_IN1                  |
|  | 001: XTLF                     |
|  | 010: UART4_RX                 |
|  | 011: UART5_RX                 |
|  | 100: RTCSEC                   |
|  | 101~111: RFU                  |
|  | ET4                           |
|  | 000: ET4_IN1                  |
|  | 001: XTLF                     |
|  | 010: UART_RX2                 |
|  | 011: UART_RX0                 |
|  | 100: CMP1O                    |
|  | 101: CMP2O                    |
|  | 110: RTCSEC                   |
|  | 111: LPTO                     |
|  | ETx 预分频寄存器 1:                 |
|  | 00~FF, 00 表示不分频, FF 表示 256 分频 |
|  | ETx 预分频寄存器 2:                 |
|  | 00~FF, 00 表示不分频, FF 表示 256 分频 |
|  | 控制类参数:                        |
|  | 引脚输入数字滤波使能:                   |
|  | DISABLE: 禁止引脚输入数字滤波           |
|  | ENABLE: 使能引脚输入数字滤波            |
|  | 工作模式选择:                       |
|  | 0: 定时/计数模式                    |
|  | 1: 捕捉模式                       |
|  | 扩展定时器级联使能:                    |
|  | DISABLE: 禁止扩展定时器级联            |
|  | ENABLE: 使能扩展定时器级联             |
|  | 计数模式采沿方式选择:                   |
|  | 0: 上升沿采样                      |
|  | 1: 下降沿采样                      |
|  | PWM 输出控制:                     |
|  | DISABLE: 禁止 PWM 输出            |
|  | ENABLE: 使能 PWM 输出             |
|  | 捕捉模式控制:                       |
|  |                               |
|  |                               |
|  |                               |



|      |  |
|------|--|
|      | 0: 脉冲周期捕捉<br>1: 脉宽捕捉<br>带清零捕捉模式控制:<br>0: 捕捉不清零<br>1: 时间触发捕捉, 使能后计数器保持 0, 捕捉到第一个有消沿之后 timer 才开始计数<br>单次捕捉控制:<br>0: 连续捕捉<br>1: 单次捕捉有效, 在捕捉到一次脉冲周期后计数器停止, 若再需要捕捉需重新启动<br>捕捉沿选择:<br>0: 上升沿捕捉<br>1: 下降沿捕捉<br>ETx 初值寄存器: ETx 初值<br>ETx 比较寄存器: ETx 比较值<br>扩展定时器比较中断使能:<br>DISABLE: 禁止扩展定时器比较中断<br>ENABLE: 使能扩展定时器比较中断<br>扩展定时器捕捉中断使能:<br>DISABLE: 禁止扩展定时器捕捉中断<br>ENABLE: 使能扩展定时器捕捉中断<br>扩展定时器溢出中断使能:<br>DISABLE: 禁止扩展定时器溢出中断<br>ENABLE: 使能扩展定时器溢出中断 |
| 输出参数 | 无  |
| 返回值  | 无  |

**ETIMx\_Deinit**

|            |                                      |
|------------|--------------------------------------|
| <b>函数名</b> | <b>ETIMx_Deinit</b>                  |
| 函数原型       | void ETIMx_Deinit(ETIMx_Type* ETIMx) |
| 功能描述       | ET 初始化为默认值                           |
| 输入参数 1     | 定时器名称: ETIM1/ETIM2/ETIM3/ETIM4       |
| 输出参数       | 无                                    |
| 返回值        | 无                                    |

**ETIMx\_ETxCR\_CEN\_Setable**

|               |   |
|---------------|---|
| <b>函数名</b>    | <b>ETIMx_ETxCR_CEN_Setable</b>                                      |
| <b>函数原型</b>   | void ETIMx_ETxCR_CEN_Setable(ETIMx_Type* ETIMx , FunState NewState) |
| <b>功能描述</b>   | 配置 ET 计数器启动   |
| <b>输入参数 1</b> | 定时器名称: ETIM1/ETIM2/ETIM3/ETIM4                                      |
| <b>输入参数 2</b> | DISABLE: 停止定时器<br>ENABLE: 启动定时器                                     |
| <b>输出参数</b>   | 无   |
| <b>返回值</b>    | 无   |

**ETIMx\_ETxCR\_CEN\_Getable**

|               |   |
|---------------|---|
| <b>函数名</b>    | <b>ETIMx_ETxCR_CEN_Getable</b>                      |
| <b>函数原型</b>   | FunState ETIMx_ETxCR_CEN_Getable(ETIMx_Type* ETIMx) |
| <b>功能描述</b>   | 获取 ET 计数器启动状态                                       |
| <b>输入参数 1</b> | 定时器名称: ETIM1/ETIM2/ETIM3/ETIM4                      |
| <b>输出参数</b>   | 无   |
| <b>返回值</b>    | DISABLE: 停止定时器<br>ENABLE: 启动定时器                     |

**ETIMx\_ETxCR\_MOD\_Set**

|               |   |
|---------------|---|
| <b>函数名</b>    | <b>ETIMx_ETxCR_MOD_Set</b>  |
| <b>函数原型</b>   | void ETIMx_ETxCR_MOD_Set(ETIMx_Type* ETIMx, uint32_t SetValue)                |
| <b>功能描述</b>   | 配置 ET 计数器工作模式   |
| <b>输入参数 1</b> | 定时器名称: ETIM1/ETIM2/ETIM3/ETIM4  |
| <b>输入参数 2</b> | ET 计数器工作模式:<br>ETIMx_ETxCR_MOD_COUNTER: 计数模式<br>ETIMx_ETxCR_MOD_CAPTURE: 捕捉模式 |
| <b>输出参数</b>   | 无   |
| <b>返回值</b>    | 无   |

**ETIMx\_ETxCR\_MOD\_Get**

|               |   |
|---------------|---|
| <b>函数名</b>    | <b>ETIMx_ETxCR_MOD_Get</b>  |
| <b>函数原型</b>   | uint32_t ETIMx_ETxCR_MOD_Get(ETIMx_Type* ETIMx)                               |
| <b>功能描述</b>   | 获取 ET 计数器工作模式   |
| <b>输入参数 1</b> | 定时器名称: ETIM1/ETIM2/ETIM3/ETIM4  |
| <b>输出参数</b>   | 无   |
| <b>返回值</b>    | ET 计数器工作模式:<br>ETIMx_ETxCR_MOD_COUNTER: 计数模式<br>ETIMx_ETxCR_MOD_CAPTURE: 捕捉模式 |

**ETIMx\_ETxCR\_EDGESEL\_Set**

|               |  |
|---------------|--|
| <b>函数名</b>    | <b>ETIMx_ETxCR_EDGESEL_Set</b>   |
| <b>函数原型</b>   | void ETIMx_ETxCR_EDGESEL_Set(ETIMx_Type* ETIMx , uint32_t SetValue)                              |
| <b>功能描述</b>   | 配置 ET 计数器计数模式采沿方式  |
| <b>输入参数 1</b> | 定时器名称: ETIM1/ETIM2/ETIM3/ETIM4   |
| <b>输入参数 2</b> | ET 计数器计数模式采沿方式:<br>ETIMx_ETxCR_EDGESEL_RISING: 计数模式采上升沿<br>ETIMx_ETxCR_EDGESEL_FALLING: 计数模式采下降沿 |
| <b>输出参数</b>   | 无  |
| <b>返回值</b>    | 无  |

**ETIMx\_ETxCR\_EDGESEL\_Get**

|               |   |
|---------------|---|
| <b>函数名</b>    | <b>ETIMx_ETxCR_EDGESEL_Get</b>                          |
| <b>函数原型</b>   | uint32_t ETIMx_ETxCR_EDGESEL_Get(ETIMx_Type* ETIMx)     |
| <b>功能描述</b>   | 获取 ET 计数器计数模式采沿方式                                       |
| <b>输入参数 1</b> | 定时器名称: ETIM1/ETIM2/ETIM3/ETIM4                          |
| <b>输出参数</b>   | 无   |
| <b>返回值</b>    | ET 计数器计数模式采沿方式:<br>ETIMx_ETxCR_EDGESEL_RISING: 计数模式采上升沿 |



|  |                                       |
|--|---------------------------------------|
|  | ETIMx_ETxCR_EDGESEL_FALLING: 计数模式采下降沿 |
|--|---------------------------------------|

**ETIMx\_ETxPESCALE1\_Write**

|               |   |
|---------------|---|
| <b>函数名</b>    | <b>ETIMx_ETxPESCALE1_Write</b>                                      |
| <b>函数原型</b>   | void ETIMx_ETxPESCALE1_Write(ETIMx_Type* ETIMx , uint32_t SetValue) |
| <b>功能描述</b>   | 配置 ET Signal1(计数源)的预分频  |
| <b>输入参数 1</b> | 定时器名称: ETIM1/ETIM2/ETIM3/ETIM4                                      |
| <b>输入参数 2</b> | ET Signal1(计数源)的预分频: 00-FF(00 表示 1 分频, FF 表示 256 分频)                |
| <b>输出参数</b>   | 无   |
| <b>返回值</b>    | 无   |

**ETIMx\_ETxPESCALE1\_Read**

|               |  |
|---------------|--|
| <b>函数名</b>    | <b>ETIMx_ETxPESCALE1_Read</b>                        |
| <b>函数原型</b>   | uint32_t ETIMx_ETxPESCALE1_Read(ETIMx_Type* ETIMx)   |
| <b>功能描述</b>   | 获取 ET Signal1(计数源) 的预分频                              |
| <b>输入参数 1</b> | 定时器名称: ETIM1/ETIM2/ETIM3/ETIM4                       |
| <b>输出参数</b>   | 无  |
| <b>返回值</b>    | ET Signal1(计数源)的预分频: 00-FF(00 表示 1 分频, FF 表示 256 分频) |

**ETIMx\_ETxPESCALE2\_Write**

|               |   |
|---------------|---|
| <b>函数名</b>    | <b>ETIMx_ETxPESCALE2_Write</b>                                      |
| <b>函数原型</b>   | void ETIMx_ETxPESCALE2_Write(ETIMx_Type* ETIMx , uint32_t SetValue) |
| <b>功能描述</b>   | 配置 ET Signal2(计数源)的预分频  |
| <b>输入参数 1</b> | 定时器名称: ETIM1/ETIM2/ETIM3/ETIM4                                      |
| <b>输入参数 2</b> | ET Signal2(计数源)的预分频: 00-FF(00 表示 1 分频, FF 表示 256 分频)                |
| <b>输出参数</b>   | 无   |
| <b>返回值</b>    | 无   |

**ETIMx\_ETxPESCALE2\_Read**

|               |  |
|---------------|--|
| <b>函数名</b>    | <b>ETIMx_ETxPESCALE2_Read</b>                        |
| <b>函数原型</b>   | uint32_t ETIMx_ETxPESCALE2_Read(ETIMx_Type* ETIMx)   |
| <b>功能描述</b>   | 获取 ET Signal2(计数源) 的预分频                              |
| <b>输入参数 1</b> | 定时器名称: ETIM1/ETIM2/ETIM3/ETIM4                       |
| <b>输出参数</b>   | 无  |
| <b>返回值</b>    | ET Signal2(计数源)的预分频: 00-FF(00 表示 1 分频, FF 表示 256 分频) |

**ETIMx\_ETxIVR\_Write**

|               |   |
|---------------|---|
| <b>函数名</b>    | <b>ETIMx_ETxIVR_Write</b>                                     |
| <b>函数原型</b>   | void ETIMx_ETxIVR_Write(ETIMx_Type* ETIMx, uint32_t SetValue) |
| <b>功能描述</b>   | 配置 ET 计数初值  |
| <b>输入参数 1</b> | 定时器名称: ETIM1/ETIM2/ETIM3/ETIM4                                |
| <b>输出参数 2</b> | ET 计数初值: 00-FFFF  |
| <b>输出参数</b>   | 无   |
| <b>返回值</b>    | 无   |

**ETIMx\_ETxIVR\_Read**

|               |   |
|---------------|---|
| <b>函数名</b>    | <b>ETIMx_ETxIVR_Read</b>                      |
| <b>函数原型</b>   | uint32_t ETIMx_ETxIVR_Read(ETIMx_Type* ETIMx) |
| <b>功能描述</b>   | 获取 ET 计数初值                                    |
| <b>输入参数 1</b> | 定时器名称: ETIM1/ETIM2/ETIM3/ETIM4                |
| <b>输出参数</b>   | 无   |
| <b>返回值</b>    | ET 计数初值: 00-FFFF                              |

**ETIMx\_ETxCMP\_Write**

|               |   |
|---------------|---|
| <b>函数名</b>    | <b>ETIMx_ETxCMP_Write</b>                                     |
| <b>函数原型</b>   | void ETIMx_ETxCMP_Write(ETIMx_Type* ETIMx, uint32_t SetValue) |
| <b>功能描述</b>   | 配置 ET 比较值   |
| <b>输入参数 1</b> | 定时器名称: ETIM1/ETIM2/ETIM3/ETIM4                                |



|        |                 |
|--------|-----------------|
| 输入参数 2 | ET 比较值: 00-FFFF |
| 输出参数   | 无               |
| 返回值    | 无               |

**ETIMx\_ETxCMP\_Read**

|        |   |
|--------|---|
| 函数名    | <b>ETIMx_ETxCMP_Read</b>                      |
| 函数原型   | uint32_t ETIMx_ETxCMP_Read(ETIMx_Type* ETIMx) |
| 功能描述   | 获取 ET 比较值                                     |
| 输入参数 1 | 定时器名称: ETIM1/ETIM2/ETIM3/ETIM4                |
| 输出参数   | 无   |
| 返回值    | ET 比较值: 00-FFFF                               |

**ETIMx\_ETxINSEL\_SIG1SEL\_Set**

|        |  |
|--------|--|
| 函数名    | <b>ETIMx_ETxINSEL_SIG1SEL_Set</b>  |
| 函数原型   | void ETIMx_ETxINSEL_SIG1SEL_Set(ETIMx_Type* ETIMx , uint32_t SetValue)   |
| 功能描述   | 配置 ET 内部信号 1 源选择   |
| 输入参数 1 | 定时器名称: ETIM1/ETIM2/ETIM3/ETIM4   |
| 输入参数 2 | ET 内部信号 1 源选择:<br>ETIMx_ETxINSEL_SIG1SEL_GROUP1: 扩展定时器的内部信号 1 选择 Group1<br>ETIMx_ETxINSEL_SIG1SEL_GROUP2: 扩展定时器的内部信号 1 选择 Group2 |
| 输出参数   | 无  |
| 返回值    | 无  |

**ETIMx\_ETxINSEL\_SIG1SEL\_Get**

|      |  |
|------|--|
| 函数名  | <b>ETIMx_ETxINSEL_SIG1SEL_Get</b>                      |
| 函数原型 | uint32_t ETIMx_ETxINSEL_SIG1SEL_Get(ETIMx_Type* ETIMx) |
| 功能描述 | 获取 ET 内部信号 1 源选择                                       |



|        |  |
|--------|--|
| 输入参数 1 | 定时器名称: ETIM1/ETIM2/ETIM3/ETIM4   |
| 输出参数   | 无  |
| 返回值    | ET 内部信号 1 源选择:<br>ETIMx_ETxINSEL_SIG1SEL_GROUP1: 扩展定时器的内部信号 1 选择 Group1<br>ETIMx_ETxINSEL_SIG1SEL_GROUP2: 扩展定时器的内部信号 1 选择 Group2 |

### ETIMx\_ETxINSEL\_SIG2SEL\_Set

|        |  |
|--------|--|
| 函数名    | ETIMx_ETxINSEL_SIG2SEL_Set   |
| 函数原型   | void ETIMx_ETxINSEL_SIG2SEL_Set(ETIMx_Type* ETIMx, uint32_t SetValue)  |
| 功能描述   | 配置 ET 内部信号 2 源选择   |
| 输入参数 1 | 定时器名称: ETIM1/ETIM2/ETIM3/ETIM4   |
| 输入参数 2 | ET 内部信号 2 源选择:<br>ETIMx_ETxINSEL_SIG2SEL_GROUP2: 扩展定时器 3 的内部信号 2 选择 Group2<br>ETIMx_ETxINSEL_SIG2SEL_GROUP1: 扩展定时器 3 的内部信号 2 选择 Group1 |
| 输出参数   | 无  |
| 返回值    | 无  |

### ETIMx\_ETxINSEL\_SIG2SEL\_Get

|        |   |
|--------|---|
| 函数名    | ETIMx_ETxINSEL_SIG2SEL_Get  |
| 函数原型   | uint32_t ETIMx_ETxINSEL_SIG2SEL_Get(ETIMx_Type* ETIMx)              |
| 功能描述   | 获取 ET 内部信号 2 源选择  |
| 输入参数 1 | 定时器名称: ETIM1/ETIM2/ETIM3/ETIM4                                      |
| 输出参数   | 无   |
| 返回值    | ET 内部信号 2 源选择:<br>ETIMx_ETxINSEL_SIG2SEL_GROUP2: 扩展定时器 3 的内部信号 2 选择 |



|  |   |
|--|---|
|  | Group2<br>ETIMx_ETxINSEL_SIG2SEL_GROUP1: 扩展定时器 3 的内部信号 2 选择<br>Group1 |
|--|---|

### ETIMx\_ETxCR\_CAPEDGE\_Set

|        |   |
|--------|---|
| 函数名    | ETIMx_ETxCR_CAPEDGE_Set   |
| 函数原型   | void ETIMx_ETxCR_CAPEDGE_Set(ETIMx_Type* ETIMx , uint32_t SetValue)                                 |
| 功能描述   | 配置 ET 计数器捕捉沿选择  |
| 输入参数 1 | 定时器名称: ETIM1/ETIM2/ETIM3/ETIM4  |
| 输入参数 2 | ET 计数器捕捉沿选择:<br>ETIMx_ETxCR_CAPEDGE_RISING: 周期捕捉模式时上沿捕捉<br>ETIMx_ETxCR_CAPEDGE_FALLING: 周期捕捉模式时下沿捕捉 |
| 输出参数   | 无   |
| 返回值    | 无   |

### ETIMx\_ETxCR\_CAPEDGE\_Get

|        |   |
|--------|---|
| 函数名    | ETIMx_ETxCR_CAPEDGE_Get   |
| 函数原型   | uint32_t ETIMx_ETxCR_CAPEDGE_Get(ETIMx_Type* ETIMx)   |
| 功能描述   | 获取 ET 计数器捕捉沿选择配置  |
| 输入参数 1 | 定时器名称: ETIM1/ETIM2/ETIM3/ETIM4  |
| 输出参数   | 无   |
| 返回值    | ET 计数器捕捉沿选择:<br>ETIMx_ETxCR_CAPEDGE_RISING: 周期捕捉模式时上沿捕捉<br>ETIMx_ETxCR_CAPEDGE_FALLING: 周期捕捉模式时下沿捕捉 |

### ETIMx\_ETxCR\_CAPMOD\_Set

|      |  |
|------|--|
| 函数名  | ETIMx_ETxCR_CAPMOD_Set                                   |
| 函数原型 | void ETIMx_ETxCR_CAPMOD_Set(ETIMx_Type* ETIMx , uint32_t |



|        |  |
|--------|--|
|        | SetValue)  |
| 功能描述   | 配置 ET 计数器捕捉模式  |
| 输入参数 1 | 定时器名称: ETIM1/ETIM2/ETIM3/ETIM4   |
| 输入参数 2 | ET 计数器捕捉模式:<br>ETIMx_ETxCR_CAPMOD_PERIOD: 脉冲周期捕捉<br>ETIMx_ETxCR_CAPMOD_PULSE: 脉宽捕捉 |
| 输出参数   | 无  |
| 返回值    | 无  |

### ETIMx\_ETxCR\_CAPMOD\_Get

|        |  |
|--------|--|
| 函数名    | ETIMx_ETxCR_CAPMOD_Get   |
| 函数原型   | uint32_t ETIMx_ETxCR_CAPMOD_Get(ETIMx_Type* ETIMx)                                 |
| 功能描述   | 获取 ET 计数器捕捉模式  |
| 输入参数 1 | 定时器名称: ETIM1/ETIM2/ETIM3/ETIM4   |
| 输出参数   | 无  |
| 返回值    | ET 计数器捕捉模式:<br>ETIMx_ETxCR_CAPMOD_PERIOD: 脉冲周期捕捉<br>ETIMx_ETxCR_CAPMOD_PULSE: 脉宽捕捉 |

### ETIMx\_ETxCR\_CAPONCE\_Setable

|        |  |
|--------|--|
| 函数名    | ETIMx_ETxCR_CAPONCE_Setable  |
| 函数原型   | void ETIMx_ETxCR_CAPONCE_Setable(ETIMx_Type* ETIMx, FunState NewState) |
| 功能描述   | 配置 ET 计数器单次捕捉有效  |
| 输入参数 1 | 定时器名称: ETIM1/ETIM2/ETIM3/ETIM4   |
| 输入参数 2 | ET 计数器单次捕捉有效:<br>DISABLE: 连续捕捉<br>ENABLE: 单次捕捉有效                       |
| 输出参数   | 无  |
| 返回值    | 无  |



### ETIMx\_ETxCR\_CAPONCE\_Getable

|        |   |
|--------|---|
| 函数名    | ETIMx_ETxCR_CAPONCE_Getable                             |
| 函数原型   | FunState ETIMx_ETxCR_CAPONCE_Getable(ETIMx_Type* ETIMx) |
| 功能描述   | 获取 ET 计数器单次捕捉有效配置                                       |
| 输入参数 1 | 定时器名称: ETIM1/ETIM2/ETIM3/ETIM4                          |
| 输出参数   | 无   |
| 返回值    | ET 计数器单次捕捉有效:<br>DISABLE: 连续捕捉<br>ENABLE: 单次捕捉有效        |

### ETIMx\_ETxCR\_CAPCLR\_Setable

|        |  |
|--------|--|
| 函数名    | ETIMx_ETxCR_CAPCLR_Setable   |
| 函数原型   | void ETIMx_ETxCR_CAPCLR_Setable(ETIMx_Type* ETIMx, FunState NewState)                              |
| 功能描述   | 配置 ET 计数器带清零捕捉模式   |
| 输入参数 1 | 定时器名称: ETIM1/ETIM2/ETIM3/ETIM4   |
| 输入参数 2 | ET 计数器带清零捕捉模式:<br>DISABLE: 捕捉不清零, 计数器一直自由计数<br>ENABLE: 事件触发捕捉, 使能后计数器保持 0, 捕捉到第一个有效沿之后 timer 才开始计数 |
| 输出参数   | 无  |
| 返回值    | 无  |

### ETIMx\_ETxCR\_CAPCLR\_Getable

|        |  |
|--------|--|
| 函数名    | ETIMx_ETxCR_CAPCLR_Getable                             |
| 函数原型   | FunState ETIMx_ETxCR_CAPCLR_Getable(ETIMx_Type* ETIMx) |
| 功能描述   | 获取 ET 计数器带清零捕捉模式配置                                     |
| 输入参数 1 | 定时器名称: ETIM1/ETIM2/ETIM3/ETIM4                         |



|      |   |
|------|---|
| 输出参数 | 无   |
| 返回值  | ET 计数器带清零捕捉模式：<br>DISABLE：捕捉不清零，计数器一直自由计数<br>ENABLE：事件触发捕捉，使能后计数器保持 0，捕捉到第一个有效沿之后 timer 才开始计数 |

## ETIMx\_ETxINSEL\_GRP1SEL\_Set

|        |   |
|--------|---|
| 函数名    | ETIMx_ETxINSEL_GRP1SEL_Set  |
| 函数原型   | void ETIMx_ETxINSEL_GRP1SEL_Set(ETIMx_Type* ETIMx, uint32_t SetValue)   |
| 功能描述   | 设置 ET GROUP1 信号选择   |
| 输入参数 1 | 定时器名称: ETIM1/ETIM2/ETIM3/ETIM4  |
| 输入参数 2 | ET GROUP1 信号选择:<br>ETIMx_ETxINSEL_GRP1SEL_ET1_APBCLK: APBCLK<br>ETIMx_ETxINSEL_GRP1SEL_ET1_XTLF: XTLF<br>ETIMx_ETxINSEL_GRP1SEL_ET1_RCLP: RCLP<br>ETIMx_ETxINSEL_GRP1SEL_ET1_ET1_IN0: ET1_IN0<br>ETIMx_ETxINSEL_GRP1SEL_ET2_APBCLK: APBCLK<br>ETIMx_ETxINSEL_GRP1SEL_ET2_XTLF: XTLF<br>ETIMx_ETxINSEL_GRP1SEL_ET2_RCLP: RCLP<br>ETIMx_ETxINSEL_GRP1SEL_ET2_ET2_IN0: ET2_IN0<br>ETIMx_ETxINSEL_GRP1SEL_ET3_APBCLK: APBCLK<br>ETIMx_ETxINSEL_GRP1SEL_ET3_ET3_IN0: ET3_IN0<br>ETIMx_ETxINSEL_GRP1SEL_ET3_RTCSEC: RTCSEC<br>ETIMx_ETxINSEL_GRP1SEL_ET3_RCLP: RCLP<br>ETIMx_ETxINSEL_GRP1SEL_ET4_APBCLK: APBCLK<br>ETIMx_ETxINSEL_GRP1SEL_ET4_ET4_IN0: ET4_IN0<br>ETIMx_ETxINSEL_GRP1SEL_ET4_RTC64HZ: RTC64HZ<br>ETIMx_ETxINSEL_GRP1SEL_ET4_LPTO: LPTO |
| 输出参数   | 无   |
| 返回值    | 无   |

**ETIMx\_ETxINSEL\_GRP1SEL\_Get**

|        |   |
|--------|---|
| 函数名    | ETIMx_ETxINSEL_GRP1SEL_Get  |
| 函数原型   | uint32_t ETIMx_ETxINSEL_GRP1SEL_Get(ETIMx_Type* ETIMx)  |
| 功能描述   | 获取 ET GROUP1 信号选择   |
| 输入参数 1 | 定时器名称: ETIM1/ETIM2/ETIM3/ETIM4  |
| 输出参数   | 无   |
| 返回值    | ET GROUP1 信号选择:<br>ETIMx_ETxINSEL_GRP1SEL_ET1_APBCLK: APBCLK<br>ETIMx_ETxINSEL_GRP1SEL_ET1_XTLF: XTLF<br>ETIMx_ETxINSEL_GRP1SEL_ET1_RCLP: RCLP<br>ETIMx_ETxINSEL_GRP1SEL_ET1_ET1_IN0: ET1_IN0<br>ETIMx_ETxINSEL_GRP1SEL_ET2_APBCLK: APBCLK<br>ETIMx_ETxINSEL_GRP1SEL_ET2_XTLF: XTLF<br>ETIMx_ETxINSEL_GRP1SEL_ET2_RCLP: RCLP<br>ETIMx_ETxINSEL_GRP1SEL_ET2_ET2_IN0: ET2_IN0<br>ETIMx_ETxINSEL_GRP1SEL_ET3_APBCLK: APBCLK<br>ETIMx_ETxINSEL_GRP1SEL_ET3_ET3_IN0: ET3_IN0<br>ETIMx_ETxINSEL_GRP1SEL_ET3_RTCSEC: RTCSEC<br>ETIMx_ETxINSEL_GRP1SEL_ET3_RCLP: RCLP<br>ETIMx_ETxINSEL_GRP1SEL_ET4_APBCLK: APBCLK<br>ETIMx_ETxINSEL_GRP1SEL_ET4_ET4_IN0: ET4_IN0<br>ETIMx_ETxINSEL_GRP1SEL_ET4_RTC64HZ: RTC64HZ<br>ETIMx_ETxINSEL_GRP1SEL_ET4_LPTO: LPTO |

**ETIMx\_ETxINSEL\_GRP2SEL\_Set**

|      |   |
|------|---|
| 函数名  | ETIMx_ETxINSEL_GRP2SEL_Set  |
| 函数原型 | void ETIMx_ETxINSEL_GRP2SEL_Set(ETIMx_Type* ETIMx, uint32_t SetValue) |
| 功能描述 | 设置 ET GROUP2 信号选择   |



|        |  |
|--------|--|
| 输入参数 1 | 定时器名称: ETIM1/ETIM2/ETIM3/ETIM4   |
| 输入参数 2 | ET GROUP2 信号选择:<br>ETIMx_ETxINSEL_GRP2SEL_ET1_UART0_RX: UART0_RX<br>ETIMx_ETxINSEL_GRP2SEL_ET1_UART1_RX: UART1_RX<br>ETIMx_ETxINSEL_GRP2SEL_ET1_XTLF: XTLF<br>ETIMx_ETxINSEL_GRP2SEL_ET1_ET1_IN1: ET1_IN1<br>ETIMx_ETxINSEL_GRP2SEL_ET1_ET1_IN2: ET1_IN2<br>ETIMx_ETxINSEL_GRP2SEL_ET1_CMP1O: CMP1O (比较器 1 输出)<br>ETIMx_ETxINSEL_GRP2SEL_ET1_CMP2O: CMP2O (比较器 2 输出)<br>ETIMx_ETxINSEL_GRP2SEL_ET1_LPTO: LPTO<br>ETIMx_ETxINSEL_GRP2SEL_ET2_UART2_RX: UART2_RX<br>ETIMx_ETxINSEL_GRP2SEL_ET2_UART3_RX: UART3_RX<br>ETIMx_ETxINSEL_GRP2SEL_ET2_XTLF: XTLF<br>ETIMx_ETxINSEL_GRP2SEL_ET2_ET2_IN1: ET2_IN1<br>ETIMx_ETxINSEL_GRP2SEL_ET2_ET2_IN2: ET2_IN2<br>ETIMx_ETxINSEL_GRP2SEL_ET2_CMP1O: CMP1O (比较器 1 输出)<br>ETIMx_ETxINSEL_GRP2SEL_ET2_CMP2O: CMP2O (比较器 2 输出)<br>ETIMx_ETxINSEL_GRP2SEL_ET2_LPTO: LPTO<br>ETIMx_ETxINSEL_GRP2SEL_ET3_ET3_IN12: ET3_IN1<br>ETIMx_ETxINSEL_GRP2SEL_ET3_XTLF: XTLF<br>ETIMx_ETxINSEL_GRP2SEL_ET3_UART4_RX: UART4_RX<br>ETIMx_ETxINSEL_GRP2SEL_ET3_UART5_RX: UART5_RX<br>ETIMx_ETxINSEL_GRP2SEL_ET3_RTCSEC: RTCSEC<br>ETIMx_ETxINSEL_GRP2SEL_ET4_ET4_IN1: ET4_IN1<br>ETIMx_ETxINSEL_GRP2SEL_ET4_XTLF: XTLF<br>ETIMx_ETxINSEL_GRP2SEL_ET4_UART_RX22: UART_RX2<br>ETIMx_ETxINSEL_GRP2SEL_ET4_UART_RX0: UART_RX0<br>ETIMx_ETxINSEL_GRP2SEL_ET4_CMP1O: CMP1O (比较器 1 输出)<br>ETIMx_ETxINSEL_GRP2SEL_ET4_CMP2O: CMP2O (比较器 2 输出)<br>ETIMx_ETxINSEL_GRP2SEL_ET4_RTCSEC: RTCSEC<br>ETIMx_ETxINSEL_GRP2SEL_ET4_LPTO: LPTO |
| 输出参数   | 无  |



|     |   |
|-----|---|
| 返回值 | 无 |
|-----|---|

### ETIMx\_ETxINSEL\_GRP2SEL\_Get

|        |  |
|--------|--|
| 函数名    | ETIMx_ETxINSEL_GRP2SEL_Get   |
| 函数原型   | uint32_t ETIMx_ETxINSEL_GRP2SEL_Get(ETIMx_Type* ETIMx)   |
| 功能描述   | 获取 ET GROUP2 信号选择  |
| 输入参数 1 | 定时器名称: ETIM1/ETIM2/ETIM3/ETIM4   |
| 输出参数   | 无  |
| 返回值    | ET GROUP2 信号选择:<br>ETIMx_ETxINSEL_GRP2SEL_ET1_UART0_RX: UART0_RX<br>ETIMx_ETxINSEL_GRP2SEL_ET1_UART1_RX: UART1_RX<br>ETIMx_ETxINSEL_GRP2SEL_ET1_XTLF: XTLF<br>ETIMx_ETxINSEL_GRP2SEL_ET1_ET1_IN1: ET1_IN1<br>ETIMx_ETxINSEL_GRP2SEL_ET1_ET1_IN2: ET1_IN2<br>ETIMx_ETxINSEL_GRP2SEL_ET1_CMP1O: CMP1O (比较器 1 输出)<br>ETIMx_ETxINSEL_GRP2SEL_ET1_CMP2O: CMP2O (比较器 2 输出)<br>ETIMx_ETxINSEL_GRP2SEL_ET1_LPTO: LPTO<br>ETIMx_ETxINSEL_GRP2SEL_ET2_UART2_RX: UART2_RX<br>ETIMx_ETxINSEL_GRP2SEL_ET2_UART3_RX: UART3_RX<br>ETIMx_ETxINSEL_GRP2SEL_ET2_XTLF: XTLF<br>ETIMx_ETxINSEL_GRP2SEL_ET2_ET2_IN1: ET2_IN1<br>ETIMx_ETxINSEL_GRP2SEL_ET2_ET2_IN2: ET2_IN2<br>ETIMx_ETxINSEL_GRP2SEL_ET2_CMP1O: CMP1O (比较器 1 输出)<br>ETIMx_ETxINSEL_GRP2SEL_ET2_CMP2O: CMP2O (比较器 2 输出)<br>ETIMx_ETxINSEL_GRP2SEL_ET2_LPTO: LPTO<br>ETIMx_ETxINSEL_GRP2SEL_ET3_ET3_IN12: ET3_IN1<br>ETIMx_ETxINSEL_GRP2SEL_ET3_XTLF: XTLF<br>ETIMx_ETxINSEL_GRP2SEL_ET3_UART4_RX: UART4_RX<br>ETIMx_ETxINSEL_GRP2SEL_ET3_UART5_RX: UART5_RX<br>ETIMx_ETxINSEL_GRP2SEL_ET3_RTCSEC: RTCSEC<br>ETIMx_ETxINSEL_GRP2SEL_ET4_ET4_IN1: ET4_IN1 |



|  |  |
|--|--|
|  | ETIMx_ETxINSEL_GRP2SEL_ET4_XTLF: XTLF<br>ETIMx_ETxINSEL_GRP2SEL_ET4_UART_RX22: UART_RX2<br>ETIMx_ETxINSEL_GRP2SEL_ET4_UART_RX0: UART_RX0<br>ETIMx_ETxINSEL_GRP2SEL_ET4_CMP1O: CMP1O (比较器 1 输出)<br>ETIMx_ETxINSEL_GRP2SEL_ET4_CMP2O: CMP2O (比较器 2 输出)<br>ETIMx_ETxINSEL_GRP2SEL_ET4_RTCSEC: RTCSEC<br>ETIMx_ETxINSEL_GRP2SEL_ET4_LPTO: LPTO |
|--|--|

### ETIMx\_ETxIE\_CAPIE\_Setable

| 函数名    | ETIMx_ETxIE_CAPIE_Setable  |
|--------|--|
| 函数原型   | void ETIMx_ETxIE_CAPIE_Setable(ETIMx_Type* ETIMx, FunState NewState) |
| 功能描述   | 配置 ET 捕捉中断   |
| 输入参数 1 | 定时器名称: ETIM1/ETIM2/ETIM3/ETIM4                                       |
| 输入参数 2 | ET 捕捉中断:<br>DISABLE: 禁止捕捉中断<br>ENABLE: 使能捕捉中断                        |
| 输出参数   | 无  |
| 返回值    | 无  |

### ETIMx\_ETxIE\_CAPIE\_Getable

| 函数名    | ETIMx_ETxIE_CAPIE_Getable                             |
|--------|---|
| 函数原型   | FunState ETIMx_ETxIE_CAPIE_Getable(ETIMx_Type* ETIMx) |
| 功能描述   | 获取 ET 捕捉中断  |
| 输入参数 1 | 定时器名称: ETIM1/ETIM2/ETIM3/ETIM4                        |
| 输出参数   | 无   |
| 返回值    | ET 捕捉中断:<br>DISABLE: 禁止捕捉中断<br>ENABLE: 使能捕捉中断         |

**ETIMx\_ETxIE\_OVIE\_Setable**

|        |  |
|--------|--|
| 函数名    | ETIMx_ETxIE_OVIE_Setable   |
| 函数原型   | void ETIMx_ETxIE_OVIE_Setable(ETIMx_Type* ETIMx , FunState NewState) |
| 功能描述   | 配置 ET 溢出中断   |
| 输入参数 1 | 定时器名称: ETIM1/ETIM2/ETIM3/ETIM4                                       |
| 输入参数 2 | ET 溢出中断:<br>DISABLE: 禁止溢出中断<br>ENABLE: 使能溢出中断                        |
| 输出参数   | 无  |
| 返回值    | 无  |

**ETIMx\_ETxIE\_OVIE\_Getable**

|        |  |
|--------|--|
| 函数名    | ETIMx_ETxIE_OVIE_Getable                             |
| 函数原型   | FunState ETIMx_ETxIE_OVIE_Getable(ETIMx_Type* ETIMx) |
| 功能描述   | 获取 ET 溢出中断   |
| 输入参数 1 | 定时器名称: ETIM1/ETIM2/ETIM3/ETIM4                       |
| 输出参数   | 无  |
| 返回值    | ET 溢出中断:<br>DISABLE: 禁止溢出中断<br>ENABLE: 使能溢出中断        |

|     |   |
|-----|---|
| 返回值 | ET 比较中断标志:<br>RESET: 未发生比较中断<br>SET: 发生比较中断 |
|-----|---|

**ETIMx\_ETxIF\_CAPIF\_Clr**

|      |   |
|------|---|
| 函数名  | ETIMx_ETxIF_CAPIF_Clr                         |
| 函数原型 | void ETIMx_ETxIF_CAPIF_Clr(ETIMx_Type* ETIMx) |



|        |                                |
|--------|--------------------------------|
| 功能描述   | 清除 ET 产生捕捉信号标志                 |
| 输入参数 1 | 定时器名称: ETIM1/ETIM2/ETIM3/ETIM4 |
| 输出参数   | 无                              |
| 返回值    | 无                              |

### ETIMx\_ETxIF\_CAPIF\_Chk

|        |   |
|--------|---|
| 函数名    | ETIMx_ETxIF_CAPIF_Chk                               |
| 函数原型   | FlagStatus ETIMx_ETxIF_CAPIF_Chk(ETIMx_Type* ETIMx) |
| 功能描述   | 获取 ET 产生捕捉信号标志                                      |
| 输入参数 1 | 定时器名称: ETIM1/ETIM2/ETIM3/ETIM4                      |
| 输出参数   | 无   |
| 返回值    | ET 产生捕捉信号标志:<br>RESET: 未捕捉到指定的沿<br>SET: 捕捉到指定的沿     |

### ETIMx\_ETxIF\_OVIF\_Clr

|        |  |
|--------|--|
| 函数名    | ETIMx_ETxIF_OVIF_Clr                         |
| 函数原型   | void ETIMx_ETxIF_OVIF_Clr(ETIMx_Type* ETIMx) |
| 功能描述   | 清除 ET 产生溢出信号标志                               |
| 输入参数 1 | 定时器名称: ETIM1/ETIM2/ETIM3/ETIM4               |
| 输出参数   | 无  |
| 返回值    | 无  |

### ETIMx\_ETxIF\_OVIF\_Chk

|        |  |
|--------|--|
| 函数名    | ETIMx_ETxIF_OVIF_Chk                               |
| 函数原型   | FlagStatus ETIMx_ETxIF_OVIF_Chk(ETIMx_Type* ETIMx) |
| 功能描述   | 获取 ET 产生溢出信号标志                                     |
| 输入参数 1 | 定时器名称: ETIM1/ETIM2/ETIM3/ETIM4                     |
| 输出参数   | 无  |



|     |   |
|-----|---|
| 返回值 | ET 产生溢出信号标志：<br>RESET：未产生溢出<br>SET：产生计数溢出 |
|-----|---|

### ETIMx\_ETxIF\_EDGESTA\_Chk

|        |   |
|--------|---|
| 函数名    | ETIMx_ETxIF_EDGESTA_Chk                                     |
| 函数原型   | FlagStatus ETIMx_ETxIF_EDGESTA_Chk(ETIMx_Type* ETIMx)       |
| 功能描述   | 获取 ET 捕捉沿状态   |
| 输入参数 1 | 定时器名称：ETIM1/ETIM2/ETIM3/ETIM4                               |
| 输出参数   | 无   |
| 返回值    | ET 捕捉沿状态：<br>RESET：脉冲宽度捕捉模式时表示捕捉到上沿<br>SET：脉冲宽度捕捉模式时表示捕捉到下沿 |

### ETIMx\_ETxCR\_EXFLT\_Setable

|        |  |
|--------|--|
| 函数名    | ETIMx_ETxCR_EXFLT_Setable  |
| 函数原型   | void ETIMx_ETxCR_EXFLT_Setable(ETIMx_Type* ETIMx, FunState NewState) |
| 功能描述   | ET 计数器外部输入引脚数字滤波功能配置   |
| 输入参数 1 | 定时器名称：ETIM1/ETIM2/ETIM3/ETIM4  |
| 输入参数 2 | ET 计数器外部输入引脚数字滤波功能使能：<br>DISABLE：禁止引脚输入信号数字滤波<br>ENABLE：使能引脚输入信号数字滤波 |
| 输出参数   | 无  |
| 返回值    | 无  |

### ETIMx\_ETxCR\_EXFLT\_Getable

|      |   |
|------|---|
| 函数名  | ETIMx_ETxCR_EXFLT_Getable                             |
| 函数原型 | FunState ETIMx_ETxCR_EXFLT_Getable(ETIMx_Type* ETIMx) |



|        |  |
|--------|--|
| 功能描述   | 获取 ET 计数器外部输入引脚数字滤波功能配置  |
| 输入参数 1 | 定时器名称: ETIM1/ETIM2/ETIM3/ETIM4   |
| 输出参数   | 无  |
| 返回值    | ET 计数器外部输入引脚数字滤波功能使能:<br>DISABLE: 禁止引脚输入信号数字滤波<br>ENABLE: 使能引脚输入信号数字滤波 |

### ETIMx\_ETxCNT\_Read

|        |   |
|--------|---|
| 函数名    | ETIMx_ETxCNT_Read                             |
| 函数原型   | uint32_t ETIMx_ETxCNT_Read(ETIMx_Type* ETIMx) |
| 功能描述   | 获取 ET 计数器数值                                   |
| 输入参数 1 | 定时器名称: ETIM1/ETIM2/ETIM3/ETIM4                |
| 输出参数   | 无   |
| 返回值    | ET 计数器数值: 0-FFFF                              |

## 2.3 ADC 库函数

| 序号 | 函数名                         | 描述              |
|----|-----------------------------|-----------------|
| 1  | ANAC_Deinit                 | ANAC 初始化为缺省值    |
| 2  | ANAC_PDRCON_PDRCFG_Set      | 设置 PDR 下电复位电压   |
| 3  | ANAC_PDRCON_PDRCFG_Get      | 获取 PDR 下电复位电压   |
| 4  | ANAC_PDRCON_PDREN_Setable   | 设置 PDR 下电复位使能   |
| 5  | ANAC_PDRCON_PDREN_Getable   | 获取 PDR 下电复位使能状态 |
| 6  | ANAC_BORCON_BOR_PDRCFG_Set  | 设置 BOR 下电复位电压   |
| 7  | ANAC_BORCON_BOR_PDRCFG_Get  | 获取 BOR 下电复位电压   |
| 8  | ANAC_BORCON_OFF_BOR_Setable | 设置 BOR 下电复位使能   |
| 9  | ANAC_BORCON_OFF_BOR_Getable | 获取 BOR 下电复位使能状态 |
| 10 | ANAC_SVD_Init               | SVD 初始化设置       |
| 11 | ANAC_SVDCON_SVDEN_Setable   | 设置 SVD 使能       |
| 12 | ANAC_SVDCON_SVDEN_Getable   | 获取 SVD 使能状态     |
| 13 | ANAC_SVDCFG_SVDMOD_Set      | 设置 SVD 工作模式     |



|    |                             |                       |
|----|-----------------------------|-----------------------|
| 14 | ANAC_SVDCFG_SVDMOD_Get      | 获取 SVD 工作模式           |
| 15 | ANAC_SVDCFG_PFIE_Setable    | 设置 SVD 电源跌落中断使能       |
| 16 | ANAC_SVDCFG_PFIE_Getable    | 获取 SVD 电源跌落中断使能状态     |
| 17 | ANAC_SVDCFG_PRIIE_Setable   | 设置 SVD 电源恢复中断使能       |
| 18 | ANAC_SVDCFG_PRIIE_Getable   | 获取 SVD 电源恢复中断使能状态     |
| 19 | ANAC_SVDCFG_SVDLVL_Set      | 设置 SVD 报警阈值           |
| 20 | ANAC_SVDCFG_SVDLVL_Get      | 获取 SVD 报警阈值           |
| 21 | ANAC_SVDCFG_DFEN_Setable    | 设置 SVD 数字滤波使能         |
| 22 | ANAC_SVDCFG_DFEN_Getable    | 获取 SVD 数字滤波使能状态       |
| 23 | ANAC_SVDCFG_SVDITVL_Set     | 设置 SVD 间歇使能间隔         |
| 24 | ANAC_SVDCFG_SVDITVL_Get     | 获取 SVD 间歇使能间隔         |
| 25 | ANAC_SVDCON_SVDTE_Setable   | 设置 SVD 测试使能           |
| 26 | ANAC_SVDCON_SVDTE_Getable   | 获取 SVD 测试使能状态         |
| 27 | ANAC_SVDSIF_SVDO_Chk        | 获取 SVD 电源检测输出状态       |
| 28 | ANAC_SVDSIF_PFF_Clr         | 清除 SVD 电源跌落中断标志       |
| 29 | ANAC_SVDSIF_PFF_Chk         | 获取 SVD 电源跌落中断标志状态     |
| 30 | ANAC_SVDSIF_PRF_Clr         | 清除 SVD 电源恢复中断标志       |
| 31 | ANAC_SVDSIF_PRF_Chk         | 获取 SVD 电源恢复中断标志状态     |
| 32 | ANAC_SVDSIF_SVDR_Chk        | 获取 SVD 内部滤波后的电压检测标志状态 |
| 33 | ANAC_SVDVOL_CFG             | 设置 SVD 基准输入电压         |
| 34 | ANAC_SVDVOL_Get             | 获取 SVD 基准输入电压         |
| 35 | ANAC_FDETIF_FDETO_Chk       | 获取停振检测模块输出状态          |
| 36 | ANAC_FDETIE_FDET_IE_Setable | 设置 XTLF 停振检测报警中断使能    |
| 37 | ANAC_FDETIE_FDET_IE_Getable | 获取 XTLF 停振检测报警中断使能状态  |



|    |                               |                                 |
|----|-------------------------------|---------------------------------|
| 38 | ANAC_FDETIF_FDETIF_Clr        | 清除停振检测中断标志                      |
| 39 | ANAC_FDETIF_FDETIF_Chk        | 获取停振检测中断标志状态                    |
| 40 | ANAC_ADC_Init                 | ADC 初始化                         |
| 41 | ANAC_ADCCON_ADC_EN_Setable    | 设置 ADC 使能                       |
| 42 | ANAC_ADCCON_ADC_EN_Getable    | 获取 ADC 使能状态                     |
| 43 | ANAC_ADCCON_ADC_IE_Setable    | 设置 ADC 中断使能                     |
| 44 | ANAC_ADCCON_ADC_IE_Getable    | 获得 ADC 中断使能状态                   |
| 45 | ANAC_ADCCON_ADC_VANA_EN_Set   | 设置 ADC 通道选择                     |
| 46 | ANAC_ADCCON_ADC_VANA_EN_Get   | 获取 ADC 通道选择                     |
| 47 | ANAC_ADCINSEL_BUFSEL_Set      | 设置 ADC 输入通道选择                   |
| 48 | ANAC_ADCINSEL_BUFSEL_Get      | 获取 ADC 输入通道选择                   |
| 49 | ANAC_ADCTRIM_Write            | 写入 ADC 调校值                      |
| 50 | ANAC_ADCTRIM_Read             | 读取 ADC 调校值                      |
| 51 | ANAC_ADCDATA_Read             | 读取 ADC 输出数据                     |
| 52 | ANAC_ADCIF_ADC_IF_Clr         | 清除 ADC 转换完成中断标志                 |
| 53 | ANAC_ADCIF_ADC_IF_Chk         | 获取 ADC 转换完成中断标志状态               |
| 54 | ANAC_ADCIF_ADC_DONE_Chk       | 获取 ADC 转换完成输出标志                 |
| 55 | ANAC_ADCINSEL_BUFEN_Setable   | 设置 ADC 输入通道 Buffer 使能           |
| 56 | ANAC_ADCINSEL_BUFEN_Getable   | 获取 ADC 输入通道 Buffer 使能状态         |
| 57 | ANAC_ADCINSEL_BUFBY_P_Setable | 设置 ADC 输入通道 Buffer 是否 Bypass    |
| 58 | ANAC_ADCINSEL_BUFBY_P_Getable | 获取 ADC 输入通道 Buffer 是否 Bypass 状态 |
| 59 | ANAC_ADC_Channel_SetEx        | 设置 ADC 测量通道                     |
| 60 | ANAC_ADC_VoltageCalc          | ADC 电压计算                        |
| 61 | ANAC_ADC_TemperatureCalc      | ADC 温度计算                        |
| 62 | ANAC_COMPx_Init               | 比较器初始化                          |
| 63 | ANAC_COMP1CR_CMP1O_Chk        | 读取比较器 1 输出高低                    |



|    |                             |                  |
|----|-----------------------------|------------------|
| 64 | ANAC_COMP1CR_V1PSEL_Set     | 设置比较器 1 正极选择输入   |
| 65 | ANAC_COMP1CR_V1PSEL_Get     | 获取比较器 1 正极选择输入   |
| 66 | ANAC_COMP1CR_V1NSEL_Set     | 设置比较器 1 负极选择输入   |
| 67 | ANAC_COMP1CR_V1NSEL_Get     | 获取比较器 1 负极选择输入   |
| 68 | ANAC_COMP1CR_CMP1EN_Setable | 设置比较器 1 使能       |
| 69 | ANAC_COMP1CR_CMP1EN_Getable | 获取比较器 1 使能状态     |
| 70 | ANAC_COMP1CR_CMP1DF_Setable | 设置比较器 1 数字滤波使能   |
| 71 | ANAC_COMP1CR_CMP1DF_Getable | 获取比较器 1 数字滤波使能状态 |
| 72 | ANAC_COMP1CR_CMP1SEL_Set    | 设置比较器 1 中断源选择    |
| 73 | ANAC_COMP1CR_CMP1SEL_Get    | 获取比较器 1 中断源选择    |
| 74 | ANAC_COMP1CR_CMP1IE_Setable | 设置比较器 1 中断使能     |
| 75 | ANAC_COMP1CR_CMP1IE_Getable | 获取比较器 1 中断使能状态   |
| 76 | ANAC_COMP1CR_CMP1IF_Clr     | 清除比较器 1 中断标志     |
| 77 | ANAC_COMP1CR_CMP1IF_Chk     | 获取比较器 1 中断标志状态   |
| 78 | ANAC_COMP2CR_CMP2O_Chk      | 读取比较器 2 输出高低     |
| 79 | ANAC_COMP2CR_V2PSEL_Set     | 设置比较器 2 正极选择输入   |
| 80 | ANAC_COMP2CR_V2PSEL_Get     | 获取比较器 2 正极选择输入   |
| 81 | ANAC_COMP2CR_V2NSEL_Set     | 设置比较器 2 负极选择输入   |
| 82 | ANAC_COMP2CR_V2NSEL_Get     | 获取比较器 2 负极选择输入   |
| 83 | ANAC_COMP2CR_CMP2EN_Setable | 设置比较器 2 使能       |
| 84 | ANAC_COMP2CR_CMP2EN_Getable | 获取比较器 2 使能状态     |
| 85 | ANAC_COMP2CR_CMP2DF_Setable | 设置比较器 2 数字滤波使能   |
| 86 | ANAC_COMP2CR_CMP2DF_Getable | 获取比较器 2 数字滤波使能状态 |
| 87 | ANAC_COMP2CR_CMP2SEL_Set    | 设置比较器 2 中断源选择    |
| 88 | ANAC_COMP2CR_CMP2SEL_Get    | 获取比较器 2 中断源选择    |
| 89 | ANAC_COMP2CR_CMP2IE_Setable | 设置比较器 2 中断使能     |
| 90 | ANAC_COMP2CR_CMP2IE_Getable | 获取比较器 2 中断使能状态   |
| 91 | ANAC_COMP2CR_CMP2IF_Clr     | 清除比较器 2 中断标志     |
| 92 | ANAC_COMP2CR_CMP2IF_Chk     | 获取比较器 2 中断标志状态   |



|    |                             |                          |
|----|-----------------------------|--------------------------|
| 93 | ANAC_COMPICR_BUFBYP_Setable | 设置比较器 Buffer Bypass 使能   |
| 94 | ANAC_COMPICR_BUFBYP_Getable | 获取比较器 Buffer Bypass 使能状态 |
| 95 | ANAC_COMPICR_BUFENB_Setable | 设置比较器 Buffer 使能          |
| 96 | ANAC_COMPICR_BUFENB_Getable | 获取比较器 Buffer 使能状态        |
| 97 | ANAC_TRNGCON_TRNGEN_Setable | 设置 TRNG 使能               |
| 98 | ANAC_TRNGCON_TRNGEN_Getable | 获取 TRNG 使能状态             |

### ANAC\_Deinit

| 函数名    | ANAC_Deinit            |
|--------|------------------------|
| 函数原型   | void ANAC_Deinit(void) |
| 功能描述   | ANAC 初始化为缺省值           |
| 输入参数 1 | 无                      |
| 输出参数   | 无                      |
| 返回值    | 无                      |

### ANAC\_ADC\_Init

|        |  |
|--------|--|
| 函数名    | ANAC_ADC_Init  |
| 函数原型   | void ANAC_ADC_Init(ANAC_ADC_InitTypeDef* para)   |
| 功能描述   | ADC 初始化  |
| 输入参数 1 | <div>ADC 参数:</div> <div><div>ADC 调校值:</div><div>0X3FF: ADC 转电压</div><div>0X640: ADC 转温度</div></div> <div><div>ADC 内部、外部通道选择:</div><div>0: ADC 作用于温度传感器</div><div>1: ADC 作用于测量外部电压</div></div> <div><div>ADC 通道选择:</div><div>0000-0101: 保留，禁止使用</div><div>0110: VDD 主电源</div></div> |



|      |  |
|------|--|
|      | 0111: 保留<br>1000: ADC_IN1<br>1001: ADC_IN2<br>1010: ADC_IN3<br>1011: ADC_IN4<br>1100: ADC_IN5<br>1101: ADC_IN6<br>1110: ADC_IN7<br>1111: ADC_IN8<br><br>ADC 输入通道 Buffer 使能:<br>DISABLE: 禁止 ADC 输入通道 Buffer<br>ENABLE: 使能 ADC 输入通道 Buffer<br><br>ADC Buffer Bypass:<br>0: 停止 Bypass , ADC 测量外部信号<br>1: 发生 Bypass , ADC 测量电源电压<br><br>ADC 中断使能:<br>DISABLE: 禁止 ADC 中断<br>ENABLE: 使能 ADC 中断<br><br>ADC 使能:<br>DISABLE: 禁止 ADC<br>ENABLE: 使能 ADC |
| 输出参数 | 无  |
| 返回值  | 无  |

### ANAC\_ADCCON\_ADC\_EN\_Setable

|        |  |
|--------|--|
| 函数名    | ANAC_ADCCON_ADC_EN_Setable                         |
| 函数原型   | void ANAC_ADCCON_ADC_EN_Setable(FunState NewState) |
| 功能描述   | 设置 ADC 使能  |
| 输入参数 1 | ADC 使能:<br>DISABLE: 禁止 ADC<br>ENABLE: 使能 ADC       |
| 输出参数   | 无  |
| 返回值    | 无  |

**ANAC\_ADCCON\_ADC\_EN\_Getable**

|        |  |
|--------|--|
| 函数名    | <b>ANAC_ADCCON_ADC_EN_Getable</b>            |
| 函数原型   | void ANAC_ADCTRIM_Write(uint32_t SetValue)   |
| 功能描述   | 获取 ADC 使能状态                                  |
| 输入参数 1 | 无  |
| 输出参数   | 无  |
| 返回值    | ADC 使能:<br>DISABLE: 禁止 ADC<br>ENABLE: 使能 ADC |

**ANAC\_ADCCON\_ADC\_IE\_Setable**

|        |  |
|--------|--|
| 函数名    | <b>ANAC_ADCCON_ADC_IE_Setable</b>                    |
| 函数原型   | void ANAC_ADCCON_ADC_IE_Setable(FunState NewState)   |
| 功能描述   | 设置 ADC 中断使能  |
| 输入参数 1 | ADC 中断使能:<br>DISABLE: 禁止 ADC 中断<br>ENABLE: 使能 ADC 中断 |
| 输出参数   | 无  |
| 返回值    | 无  |

**ANAC\_ADCCON\_ADC\_IE\_Getable**

|        |   |
|--------|---|
| 函数名    | <b>ANAC_ADCCON_ADC_IE_Getable</b>         |
| 函数原型   | FunState ANAC_ADCCON_ADC_IE_Getable(void) |
| 功能描述   | 获得 ADC 中断使能状态                             |
| 输入参数 1 | 无   |
| 输出参数   | 无   |
| 返回值    | ADC 中断使能:<br>DISABLE: 禁止 ADC 中断           |



|  |                   |
|--|-------------------|
|  | ENABLE: 使能 ADC 中断 |
|--|-------------------|

### ANAC\_ADCCON\_ADC\_VANA\_EN\_Set

|        |   |
|--------|---|
| 函数名    | ANAC_ADCCON_ADC_VANA_EN_Set                           |
| 函数原型   | void ANAC_ADCCON_ADC_VANA_EN_Set(uint32_t SetValue)   |
| 功能描述   | 设置 ADC 通道选择   |
| 输入参数 1 | ADC 内部、外部通道选择:<br>0: ADC 作用于温度传感器<br>1: ADC 作用于测量外部电压 |
| 输出参数   | 无   |
| 返回值    | 无   |

### ANAC\_ADCCON\_ADC\_VANA\_EN\_Get

|        |   |
|--------|---|
| 函数名    | ANAC_ADCCON_ADC_VANA_EN_Get                           |
| 函数原型   | uint32_t ANAC_ADCCON_ADC_VANA_EN_Get(void)            |
| 功能描述   | 获取 ADC 通道选择   |
| 输入参数 1 | 无   |
| 输出参数   | 无   |
| 返回值    | ADC 内部、外部通道选择:<br>0: ADC 作用于温度传感器<br>1: ADC 作用于测量外部电压 |

### ANAC\_ADCINSEL\_BUFSEL\_Set

|        |   |
|--------|---|
| 函数名    | ANAC_ADCINSEL_BUFSEL_Set                          |
| 函数原型   | void ANAC_ADCINSEL_BUFSEL_Set(uint32_t SetValue)  |
| 功能描述   | 设置 ADC 输入通道选择                                     |
| 输入参数 1 | ADC 通道选择:<br>0000-0101: 保留, 禁止使用<br>0110: VDD 主电源 |



|      |  |
|------|--|
|      | 0111: 保留<br>1000: ADC_IN1<br>1001: ADC_IN2<br>1010: ADC_IN3<br>1011: ADC_IN4<br>1100: ADC_IN5<br>1101: ADC_IN6<br>1110: ADC_IN7<br>1111: ADC_IN8 |
| 输出参数 | 无  |
| 返回值  | 无  |

### ANAC\_ADCINSEL\_BUFSEL\_Get

|        |   |
|--------|---|
| 函数名    | ANAC_ADCINSEL_BUFSEL_Get  |
| 函数原型   | uint32_t ANAC_ADCINSEL_BUFSEL_Get(void)   |
| 功能描述   | 获取 ADC 输入通道选择   |
| 输入参数 1 | 无   |
| 输出参数   | 无   |
| 返回值    | ADC 通道选择:<br>0000-0101: 保留, 禁止使用<br>0110: VDD 主电源<br>0111: 保留<br>1000: ADC_IN1<br>1001: ADC_IN2<br>1010: ADC_IN3<br>1011: ADC_IN4<br>1100: ADC_IN5<br>1101: ADC_IN6<br>1110: ADC_IN7<br>1111: ADC_IN8 |

**ANAC\_ADCTRIM\_Write**

|        |  |
|--------|--|
| 函数名    | <b>ANAC_ADCTRIM_Write</b>                    |
| 函数原型   | void ANAC_ADCTRIM_Write(uint32_t SetValue)   |
| 功能描述   | 写入 ADC 调校值                                   |
| 输入参数 1 | ADC 调校值：<br>0X3FF: ADC 转电压<br>0X640: ADC 转温度 |
| 输出参数   | 无  |
| 返回值    | 无  |

**ANAC\_ADCTRIM\_Read**

|        |  |
|--------|--|
| 函数名    | <b>ANAC_ADCTRIM_Read</b>                     |
| 函数原型   | uint32_t ANAC_ADCTRIM_Read(void)             |
| 功能描述   | 读取 ADC 调校值                                   |
| 输入参数 1 | 无  |
| 输出参数   | 无  |
| 返回值    | ADC 调校值：<br>0X3FF: ADC 转电压<br>0X640: ADC 转温度 |

**ANAC\_ADCDATA\_Read**

|        |                                  |
|--------|----------------------------------|
| 函数名    | <b>ANAC_ADCDATA_Read</b>         |
| 函数原型   | uint32_t ANAC_ADCDATA_Read(void) |
| 功能描述   | 读取 ADC 输出数据                      |
| 输入参数 1 | 无                                |
| 输出参数   | 无                                |
| 返回值    | ADC 输出数据                         |

**ANAC\_ADCIF\_ADC\_IF\_Clr**

|        |                                  |
|--------|----------------------------------|
| 函数名    | ANAC_ADCIF_ADC_IF_Clr            |
| 函数原型   | void ANAC_ADCIF_ADC_IF_Clr(void) |
| 功能描述   | 清除 ADC 转换完成中断标志                  |
| 输入参数 1 | 无                                |
| 输出参数   | 无                                |
| 返回值    | 无                                |

**ANAC\_ADCIF\_ADC\_IF\_Chk**

|        |  |
|--------|--|
| 函数名    | ANAC_ADCIF_ADC_IF_Chk                                  |
| 函数原型   | FlagStatus ANAC_ADCIF_ADC_IF_Chk(void)                 |
| 功能描述   | 获取 ADC 转换完成中断标志状态                                      |
| 输入参数 1 | 无  |
| 输出参数   | 无  |
| 返回值    | ADC 转换完成中断标志状态：<br>0：未发生 ADC 转换完成中断<br>1：发生 ADC 转换完成中断 |

**ANAC\_ADCIF\_ADC\_DONE\_Chk**

|        |  |
|--------|--|
| 函数名    | ANAC_ADCIF_ADC_DONE_Chk                                |
| 函数原型   | FlagStatus ANAC_ADCIF_ADC_DONE_Chk(void)               |
| 功能描述   | 获取 ADC 转换完成输出标志  |
| 输入参数 1 | 无  |
| 输出参数   | 无  |
| 返回值    | ADC 转换完成输出标志：<br>RESET：未完成 ADC 转换输出<br>SET：完成 ADC 转换输出 |

**ANAC\_ADCINSEL\_BUFEN\_Setable**

|        |  |
|--------|--|
| 函数名    | <b>ANAC_ADCINSEL_BUFEN_Setable</b>   |
| 函数原型   | void ANAC_ADCINSEL_BUFEN_Setable(FunState NewState)                                |
| 功能描述   | 设置 ADC 输入通道 Buffer 使能  |
| 输入参数 1 | ADC 输入通道 Buffer 使能状态:<br>DISABLE: 禁止 ADC 输入通道 Buffer<br>ENABLE: 使能 ADC 输入通道 Buffer |
| 输出参数   | 无  |
| 返回值    | 无  |

**ANAC\_ADCINSEL\_BUFEN\_Getable**

|        |  |
|--------|--|
| 函数名    | <b>ANAC_ADCINSEL_BUFEN_Getable</b>   |
| 函数原型   | FunState ANAC_ADCINSEL_BUFEN_Getable(void)   |
| 功能描述   | 获取 ADC 输入通道 Buffer 使能状态  |
| 输入参数 1 | 无  |
| 输出参数   | 无  |
| 返回值    | ADC 输入通道 Buffer 使能状态:<br>DISABLE: 禁止 ADC 输入通道 Buffer<br>ENABLE: 使能 ADC 输入通道 Buffer |

**ANAC\_ADCINSEL\_BUFBYP\_Setable**

|        |  |
|--------|--|
| 函数名    | <b>ANAC_ADCINSEL_BUFBYP_Setable</b>  |
| 函数原型   | void ANAC_ADCINSEL_BUFBYP_Setable(FunState NewState)                                 |
| 功能描述   | 设置 ADC 输入通道 Buffer 是否 Bypass   |
| 输入参数 1 | ADC 输入通道 Buffer 是否 Bypass:<br>0: 停止 Bypass , ADC 测量外部信号<br>1: 发生 Bypass , ADC 测量电源电压 |
| 输出参数   | 无  |
| 返回值    | 无  |

**ANAC\_ADCINSEL\_BUFBYP\_Getable**

|        |  |
|--------|--|
| 函数名    | <b>ANAC_ADCINSEL_BUFBYP_Getable</b>  |
| 函数原型   | FunState ANAC_ADCINSEL_BUFBYP_Getable(void)  |
| 功能描述   | 获取 ADC 输入通道 Buffer 是否 Bypass 状态  |
| 输入参数 1 | 无  |
| 输出参数   | 无  |
| 返回值    | ADC 输入通道 Buffer 是否 Bypass:<br>0: 停止 Bypass , ADC 测量外部信号<br>1: 发生 Bypass , ADC 测量电源电压 |

**ANAC\_ADC\_Channel\_SetEx**

|        |  |
|--------|--|
| 函数名    | <b>ANAC_ADC_Channel_SetEx</b>  |
| 函数原型   | void ANAC_ADC_Channel_SetEx(uint8_t ChSel_def)   |
| 功能描述   | 设置 ADC 测量通道  |
| 输入参数 1 | ADC 测量通道:<br>0: 温度传感器<br>1: 电源电压<br>2: CH_IN1<br>3: CH_IN2<br>4: CH_IN3<br>5: CH_IN4<br>6: CH_IN5<br>7: CH_IN6<br>8: CH_IN7<br>9: CH_IN8 |
| 输出参数   | 无  |
| 返回值    | 无  |

**ANAC\_ADC\_VoltageCalc**

|        |  |
|--------|--|
| 函数名    | ANAC_ADC_VoltageCalc                       |
| 函数原型   | float ANAC_ADC_VoltageCalc(float fADCData) |
| 功能描述   | ADC 电压计算                                   |
| 输入参数 1 | AD 值                                       |
| 输出参数   | 无  |
| 返回值    | 电压   |

**ANAC\_ADC\_TemperatureCalc**

|        |  |
|--------|--|
| 函数名    | ANAC_ADC_TemperatureCalc                                     |
| 函数原型   | float ANAC_ADC_TemperatureCalc(float fADCData , uint8_t Vdd) |
| 功能描述   | ADC 温度计算   |
| 输入参数 1 | AD 值   |
| 输出参数   | 无  |
| 返回值    | 温度   |



## 版本信息

| 版本号 | 发布日期    | 更改说明     |
|-----|---------|----------|
| 1.0 | 2018.9  | 首次发布     |
|     | 2018.10 | 修改部分参数信息 |



## 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 Alexcior, 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/>