**C代码规范**

**命名**

**结构体命名**，类型命名要表达出相应的意义，尽量少使用简写，命名风格使用英文小写形式，英文单词间使用" \_ "连接。例如：

struct T\_Touch

{

char name[NAME\_MAX];

rt\_uint8\_t type;

rt\_uint8\_t flags;

...

};

用户定义的结构体以大写字母T\_开头,首字母大写。如：T\_touch;

**枚举**成员采用全大写字母，英文单词间使用" \_ "连接。

**宏**采用全大写字母，英文单词间使用" \_ "连接。例如：

#define TRUE 1 /\*\*< boolean true \*/

#define FALSE 0 /\*\*< boolean fails \*/

头文件开头，起限定编译作用的宏，开头和结尾都用双下划线，中间用单下划线连接。如下所示：

#ifndef \_\_LCD\_H\_\_

#define \_\_LCD\_H\_\_

……

#endif

**变量命名**，禁止取单个字符（如i、j、k...），建议除了要有具体含义外，还能表明其变量类型、数据类型等，但i、j、k作局部循环变量是允许的。定义全局变量时以g\_开头。如：g\_width\_name;

**函数，变量名**采用全小写的命名风格。如：get\_battery\_voltage();

**缩进**

代码采用4空格缩进的方式编写。不同编辑器的tab符号空格数不完全相同，所以用tab符缩进时，确保是4空格。

大括号采用如下的方式

if(xx)

{

i = 0;

}

for(i = 0; i < 100; i++)

{

j = i + 1;

}

即大括号总是占用一行的方式。

**空行**

1. 避免连续多个空行。

2. 函数之间要加空行。

3. return 与其它语句之前要加空行。

4. 变量声明与执行代码之间要加空行。

5. 逻辑块之间要加空行。

**空格**

1. 操作符两端用加空格

2. 函数参数之间用加空格

3. 关键字if，for，while，switch与括号之间不加空格

4. 用./→/++/--等访问成员时，不要加空格。

**变量**

1. 指针变量声明时采用如下的方式，即 \* 紧跟指针变量名

char \*name;

2. 变量定义时，一个变量占一行。

**函数声明**

1、内部函数不在.h文件声明，仅在源文件作为内部函数声明。

2、全局变量尽量在本源文件使用，涉及传参使用接口函数。

**文件头注释**

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Copyright (C), 2017-2027, TOPBAND Co., Ltd.

File name: // 文件名

Author: Version: Date: // 作者、版本及完成日期

Description: // 用于详细说明此程序文件完成的主要功能，与其他模块

// 或函数的接口，输出值、取值范围、含义及参数间的控

// 制、顺序、独立或依赖等关系

History: // 修改历史记录列表，每条修改记录应包括修改日期、修改

// 者及修改内容简述

1. Date:

Author:

Modification:

2. ...

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

**函数头部注释（备选)**

列出：函数的目的/功能、输入参数、输出参数、返回值、调用关系（函数、表）等。

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Function: // 函数名称

Description: // 函数功能、性能等的描述

Input: // 输入参数说明，包括每个参数的作

// 用、取值说明及参数间关系。

Output: // 对输出参数的说明。

Return: // 函数返回值的说明

Others: // 其它说明

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/