1. 定义一个列表和列表项

List\_t TestList;

ListItem\_t ListItem1;

1. 列表初始化

void vListInitialise(List\*const pxList)

{

pxList->pxIndex = ( ListItem\_t \* ) &( pxList->xListEnd ); (1)

pxList->xListEnd.xItemValue = portMAX\_DELAY; (2)

pxList->xListEnd.pxNext = ( ListItem\_t \* ) &( pxList->xListEnd ); (3) pxList->xListEnd.pxPrevious = ( ListItem\_t \* ) &( pxList->xListEnd ); (4)

pxList->uxNumberOfItems = ( UBaseType\_t ) 0U; (5)

listSET\_LIST\_INTEGRITY\_CHECK\_1\_VALUE( pxList ); (6)

listSET\_LIST\_INTEGRITY\_CHECK\_2\_VALUE( pxList ); (7)

}

(1)、xListEnd用来表示列表的末尾，而pxIndex表示列表项的索引号，此时列表只有一个列表项，那就是xListEnd，所以pxIndex指向xListEnd;

(2)、xListEnd 的列表项值初始化为portMAX\_DELAY，portMAX\_DELAY是个宏，在文件

portmacro.h中有定义。根据所使用的 MCU 的不同，portMAX\_DELAY值也不相同，可以为 0xffff或者0xffffffffUL，这里为0xffffffffUL。

(3)、初始化列表项xListEnd的pxNext变量，因为此时列表只有一个列表项 xListEnd，因此 pxNext只能指向自身。

(4)、同(3)一样，初始化 xListEnd的pxPrevious变量，指向 xListEnd 自身。

(5)、由于此时没有其他的列表项，因此 uxNumberOfItems 为 0，注意，这里没有算 xListEnd。

(6) 和 (7) 、初 始 化 列 表 项 中 用 于 完 整 性 检 查 字 段 ，只有宏

configUSE\_LIST\_DATA\_INTEGRITY\_CHECK\_BYTES为1的时候才有效。同样的根据所选的

MCU 不同其写入的值也不同，可以为0x5a5a或者0x5a5a5a5aUL。STM32是32位系统写入的是0x5a5a5a5aUL。

例：

vListInitialise(&TestList);

1. 列表项初始化

void vListInitialiseItem( ListItem\_t \* const pxItem )

{

pxItem->pvContainer = NULL; //初始化 pvContainer 为 NULL

//初始化用于完整性检查的变量，如果开启了这个功能的话。

listSET\_FIRST\_LIST\_ITEM\_INTEGRITY\_CHECK\_VALUE( pxItem );

listSET\_SECOND\_LIST\_ITEM\_INTEGRITY\_CHECK\_VALUE( pxItem );

}

列表项的初始化很简单，只是将列表项成员变量 pvContainer 初始化为 NULL，并且给用于完整性检查的变量赋值。有朋友可能会问，列表项的成员变量比列表要多，怎么初始化函数就这么短？其他的成员变量什么时候初始化呢？这是因为列表项要根据实际使用情况来初始化，比如任务创建函数 xTaskCreate()就会对任务堆栈中的 xStateListItem 和 xEventListItem 这两个列表项中的其他成员变量在做初始化。

例：

vListInitialiseItem(&ListItem1);

1. 插入列表项

vListInsert(&TestList,&ListItem1);

1. 删除列表项

uxListRemove(&ListItem1);

1. 在列表末尾添加列表项

vListInsertEnd(&TestList,&ListItem1);