**doxygen** 风格的注释

当别的程序或用户会使用到我们程序中的接口或数据结构时，如果在构建代码的时候就有意识的加入doxygen 风格的注释，那么当程序开发完，使用doxygen 工具就可以自动生成对你程序接口和数据结构描述的文档。因此建议在程序中使用doxygen 风格的注释来对接口函数和数据结构进行注释，而对于程序内部私有的数据结构，以及对程序逻辑性的一些注解，则采用传统的C 语言的块注释/\* \*/和行注释//就行了。

● doxygen 的注释块

|  |
| --- |
| /\*\*  \* @\*\*\* xxxxxxxxxxx  \* @\*\*\* xxxxxxxxxxxxxxxxxxxxx  \*  \* xxxxxxxxxxxxxxxxxx  \*/ |

doxygen 支持好几种注释风格，我们仅使用上面这一种即可,即在注释块开始使用两个星号，

该注释块是放在所要描述内容的上方的。

在后面我们还会看到如下所示的注释块，这种注释块是放在所要描述内容的右边的。

|  |
| --- |
| /\*\*< xxxxxxxxxxxxxxxx \*/ |

● doxygen 的常用指令

|  |
| --- |
| @file 文件名  @author 作者  @date 时间  @addtogroup 添加到一个组。  @brief 简要说明  @param 格式为 @param [in] \ [out] 用于函数说明中，in out 为该参数的类型，后面接参数名字，然后再接关于该参数的说明  @return 对函数返回值进行说明  @retval 格式为 @retval value 传回值说明主要用于函式说明中，说明特定传回值的意义。所以后面要先接一个传回值。然后在放该传回值的说明。  @note 注解信息  @attention 注意信息  @warning 警告信息  @see 表示交叉参考 |

* 文件头

|  |
| --- |
| /\*\*  \* The operating system emulation layer  \*  \* Copyright (c) 2012 Quintic .  \*  \* @file  \* @brief  \* The operating system emulation layer.  \* @author Fuquan.Zhang <fqzhang@ Quintic.cn>  \* @date 2012/07/05  \*  \*/ |

* 函数

列出函数的功能描述(@brief)，入口参数(@param[in])，出口参数(@param[out])，返回值(@return)。

|  |
| --- |
| /\*\*  \* @brief Write File operation.  \*  \* @param[in] fd - file description to an open file.  \* @param[out] buff - pointer to the buffer that containing the data to write to the file.  \* @param[in] size - number of bytes to write to the file.  \* @return - if success return actual write bytes.otherwize return -1.  \*/  Int INNO\_OS\_WriteFile(INNO\_FileHandle\_T fd, void\* buff, INNOuint32 size); |

* 结构体 或 枚举

|  |
| --- |
| /\*\*  \* @brief File Stat structure  \*/  struct INNO\_FileStat  {  INNOuint32 attrib; /\*\*< file attribute \*/ or ///< file attribute  INNOuint32 size; /\*\*< file size \*/  INNOuint32 atime; /\*\*< time of last access \*/  INNOuint32 mtime; /\*\*< time of last modified \*/  INNOuint32 ctime; /\*\*< time of create \*/  }; |

* 变量

|  |
| --- |
| /\*\* use to store platform callback functions \*/  struct INNO\_HIF\_PlatfromFuncs \* g\_client\_funs; |