



www.enjoylinux.cn

第十九课

#pragma预处理分析



版权声明：本课件及其印刷物、视频的版权归成都国嵌信息技术有限公司所有，并保留所有权力：任何单位或个人未经成都国嵌信息技术有限公司书面授权，不得使用该课件及其印刷物、视频从事商业、教学活动。已经取得书面授权的，应在授权范围内使用，并注明“来源：国嵌”。违反上述声明者，我们将追究其法律责任。

#pragma简介



- ❖ #pragma是编译器指示字，用于指示编译器完成一些特定的动作
- ❖ #pragma所定义的很多指示字是编译器和操作系统特有的
- ❖ #pragma在不同的编译器间是不可移植的
 - 预处理器将忽略它不认识的#pragma指令
 - 两个不同的编译器可能以两种不同的方式解释同一条#pragma指令

一般用法:

#pragma parameter

注：不同的parameter参数语法和意义各不相同

嵌入式Linux技术咨询QQ号: 550491596

嵌入式Linux学习交流QQ群: 65212116



#pragma message



- ❖ `message` 参数在大多数的编译器中都有相似的实现
- ❖ `message` 参数在编译时输出消息到编译输出窗口中
- ❖ `message` 可用于代码的版本控制

注意`message`是VC特有的编译器指示字，GCC中将其忽略。



嵌入式Linux技术咨询QQ号: 550491596
嵌入式Linux学习交流QQ群: 65212116



实例分析5-1



www.enjoylinux.cn



#pragma在不同编
译器下的使用示例

嵌入式Linux技术咨询QQ号: 550491596

嵌入式Linux学习交流QQ群: 65212116



#pragma pack

❖ 什么是内存对齐？

不同类型的数据在内存中按照一定的规则排列；而不是顺序的一个接一个的排放，这就是对齐。

```
struct Test1
{
    char c1;
    short s;
    char c2;
    int i;
};

struct Test2
{
    char c1;
    char c2;
    short s;
    int i;
};
```

两种类型所占的内存空间是否相同？

嵌入式Linux技术咨询QQ号: 550491596
嵌入式Linux学习交流QQ群: 65212116



#pragma pack



❖ 为什么需要内存对齐?

- CPU对内存的读取不是连续的，而是分成块读取的，块的大小只能是1、2、4、8、16字节
- 当读取操作的数据未对齐，则需要两次总线周期来访问内存，因此性能会大打折扣
- 某些硬件平台只能从规定的地址处取某些特定类型的数据，否则抛出硬件异常

嵌入式Linux技术咨询QQ号: 550491596
嵌入式Linux学习交流QQ群: 65212116



#pragma pack

❖ #pragma pack能够改变编译器的默认对齐方式

```
#pragma pack(2)
struct Test1
{
    char c1;
    short s;
    char c2;
    int i;
};
#pragma pack()
```

```
#pragma pack(4)
struct Test2
{
    char c1;
    char c2;
    short s;
    int i;
};
#pragma pack()
```

sizeof(struct Test1) = ?

sizeof(struct Test2) = ?

嵌入式Linux技术咨询QQ号: 550491596
嵌入式Linux学习交流QQ群: 65212116



#pragma pack



❖ struct占用的内存大小

- 第一个成员起始于0偏移处
- 每个成员按其类型大小和指定对齐参数n中较小的一个进行对齐
 - 偏移地址和成员占用大小均需对齐
 - 结构体成员的对齐参数为其所有成员使用的对齐参数的最大值
- 结构体总长度必须为所有对齐参数的整数倍

嵌入式Linux技术咨询QQ号: 550491596
嵌入式Linux学习交流QQ群: 65212116



实例分析5-2



www.enjoylinux.cn



Intel和微软的面试题

结构体大小的计算

嵌入式Linux技术咨询QQ号: 550491596

嵌入式Linux学习交流QQ群: 65212116



课后思考



www.enjoylinux.cn

❖ 结构体变量是否可以直接用memcmp函数进行相等判断？为什么？



嵌入式Linux技术咨询QQ号: 550491596
嵌入式Linux学习交流QQ群: 65212116

