C语言深度剖析 石虎

第一章 关键字

绪

定义和声明最重要的区别是；定义创建了对象并为这个对象分配了内存

声明没有分配内存

1.1 auto

1.2 register

1.3 static

作用一： 修饰变量。

1.静态全局变量

作用域仅限于变量被定义的文件中。其他文件即使用extern声明也没法使用他。在定义前面的代码行也不能使用它，想要使用就得在前面加extern。介意直接在文件顶端定义静态全局变量。

2.静态局部变量

在函数体内部定义的变量，只能在该函数里使用。和静态全局变量一样，总是存在内存的静态区。变量的值在函数运行结束也不会被销毁。函数下次使用仍然是这个值。



作用二： 修饰函数。

函数前加static使得函数成为静态函数。此处的static不是指存储方式，而是指函数的作用域仅限于本文件（又称为内部函数）。不用担心自己定义的函数是否会与其他文件中的函数同名。

1.4 基本数据类型



1.5 数据命名规则

标识符的一般规则：

1.命名只管可拼读，望文生义，便于记忆。

2.命名长度应符合最小长度和最大信息的原则。英文尽量不缩写。

3.标识符由多个词组成时，每个词的第一个字母大写，其余全部小写。

4.出了驱动开发的管脚命名等，尽量避免出现数字编号。

5.多文件时全局变量或函数要加范围限定符。

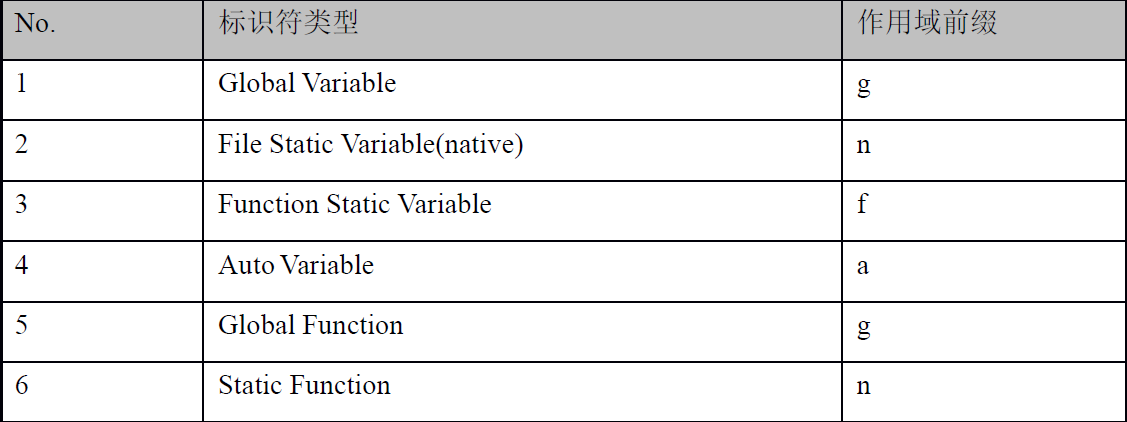
标识符的命名规则：

6.标识符分为两部分：规范标识符前缀+含义标识。非全局变量可以不使用范围限定符前缀。

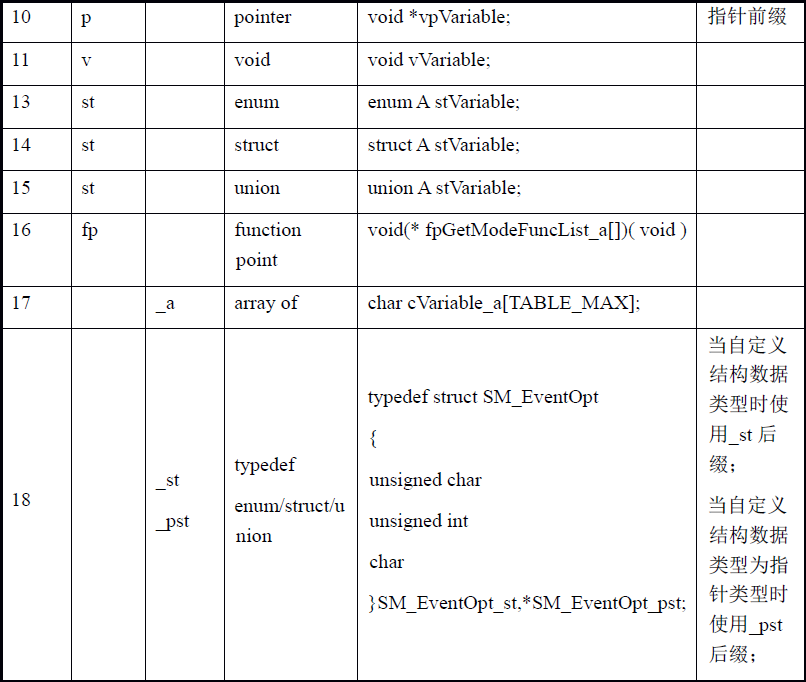
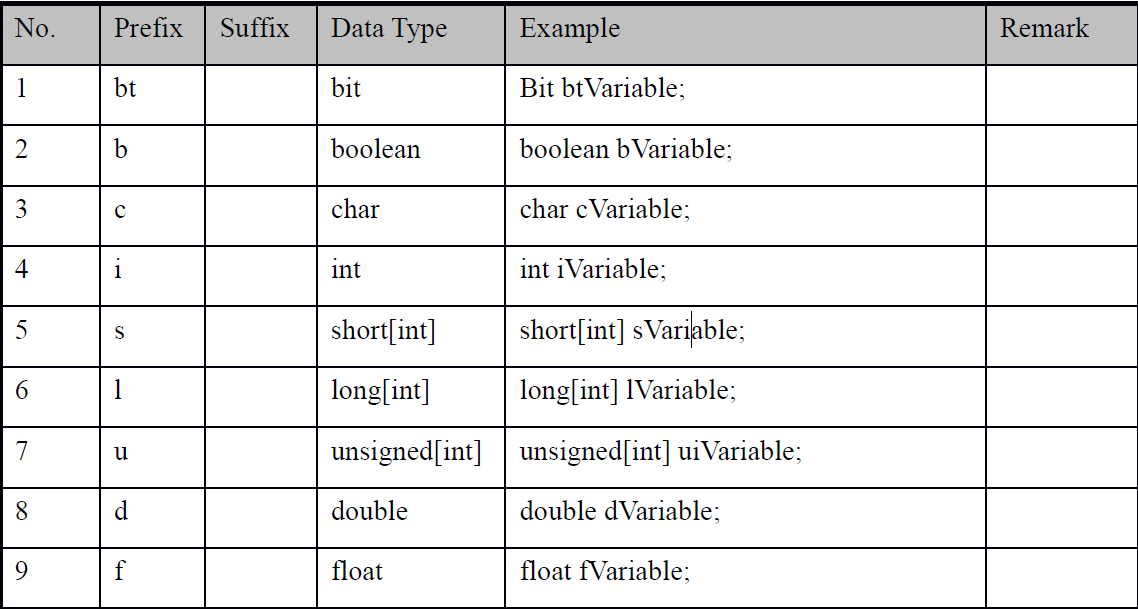
|-标识符前缀-|

模块名缩写 \_ 作用域前缀 数据类型前缀 [指针前缀] 含义标识 数组/结构后缀

作用域前缀



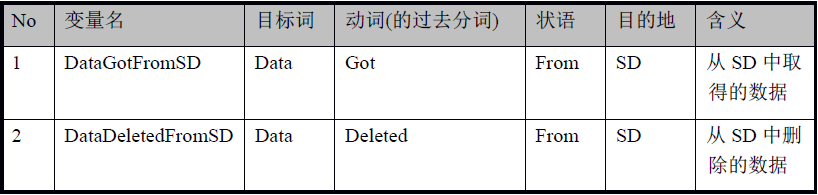
数据类型前缀



含义标识

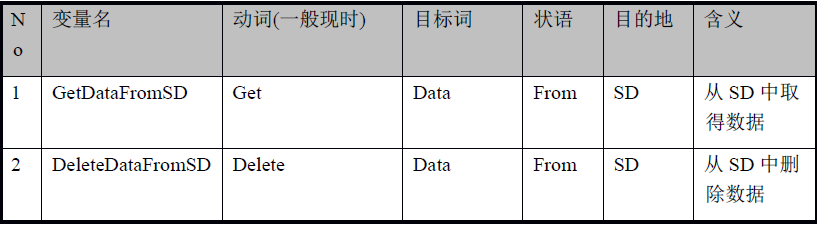
1）.变量命名使用名词性词组

变量含义标识符构成：目标词 + 动词（的过去分词）+ [状语] + [目的地]



2）.函数命名使用动词性词组

函数含义标识符构成： 动词（一般现在时）+ 目标词 + [状语] + [目的地]



7.程序中不得出现仅靠大小写区分的相似标识符。

8.一个函数名禁止被用于定义其他变量名等。

9.所有宏定义、枚举常数、只读变量全用大写字母命名，用下划线分割单词。

10.定义变量的同时千万千万别忘了初始化。定义变量时编译器并不一定清空了这块内存，它的值可能是无效数据。

11.不同类型的数据之间的运算要注意精度扩展问题。