

2.6 C 语言中的位域 1_物联网 / 嵌入式工程师 - 慕课网

“ 慕课网慕课教程 2.6 C 语言中的位域 1 涵盖海量编程基础技术教程，以图文图表的形式，把晦涩难懂的编程专业用语，以通俗易懂的方式呈现给用户。

\6. C 语言中的位域 1

有些信息在存储时，并不需要占用一个完整的字节，而只需占几个或一个二进制位。例如在存放一个开关量时，只有 0 和 1 两种状态，用 1 位二进制即可。为了节省存储空间，并使处理简便，C 语言又提供了一种数据结构，称为 "位域" 或 "位段"。

所谓 " 位域 " 是把一个字节中的二进制位划分为几个不同的区域，并说明每个区域的位数。每个域有一个域名，允许在程序中按域名进行操作。这样就可以把几个不同的对象用一个字节的二进制位域来表示。

char a = 10; ==> 0000 1010

典型的实例：

- 用 1 位二进制存放一个开关量时，只有 0 和 1 两种状态。
- 读取外部文件格式——可以读取非标准的文件格式。例如：9 位的整数。

```
struct 位域结构名
{
    类型说明符1 位域名1:位域长度1;
    类型说明符2 位域名2:位域长度2;
    类型说明符3 位域名3:位域长度3;
    .....
    类型说明符N 位域名N:位域长度M;
};
注:
类型说明符 常用int,unsigned int ,signed int,unsigned char,char这几种类型。
位域长度 位域中的数量位,要小于或等于类型的宽度
```

例如:

```
struct Bit
{
    int a:8;
    int b:2;
    int c:6;
};
```

struct Bit data;
说明位域变量data,共占用2个字节;其中,位域a占8bit,位域b占2bit,位域c占6bit

```
#include <stdio.h>
#include <string.h>
```

```
struct
{
    unsigned int width;
    unsigned int height;
} status1;
```

```
struct
{
    unsigned int width:1;
    unsigned int height:1;
} status2;
```

```
int main( )
{
    printf( "sizeof(status1) : %d\n", sizeof(status1));
```

```
printf( "sizeof(status2) : %d\n", sizeof(status2));

return 0;
}

sizeof(status1) : 8
sizeof(status2) : 4
```

说明：

上面的结构中 status2 占用了 4bytes 的空间，但是只有 2 位被用来存储值。如果您用了 32 个变量，每一个变量宽度为 1 位，那么 status 结构将使用 4 个字节，但只要您再多用一个变量，如果使用了 33 个变量，那么它将分配内存的下一段来存储第 33 个变量，这个时候就开始使用 8 个字节。

带有预定义宽度的变量被称为位域。位域可以存储多于 1 位的数，例如，需要一个变量来存储从 0 到 7 的值，您可以定义一个宽度为 3 位的位域，如下

```
struct
{
    unsigned int age : 3;
} age;
```

上面代码中 age 变量将只使用 3 位来存储这个值，如果您试图使用超过 3 位，则无法完成。

```
struct info
{
    int a:8;
    int b:2;
    int c:6;
}data;
```

data 为 info 变量，内存分配 4bytes。实际存储共占两个字节。其中位域 a 占 8 位，位域 b 占 2 位，位域 c 占 6 位。

```
#include <stdio.h>
#include <string.h>
typedef struct
{
    unsigned int age : 3;
}s_t ;

int main( )
{
    s_t t;
    t.age = 4;
    printf( "sizeof(t) : %d\n", sizeof(t) );
    printf( "t.age : %d\n", t.age );

    t.age = 7;
    printf( "t.age : %d\n", t.age );

    t.age = 8;
    printf( "t.age : %d\n", t.age );

    return 0;
}

sizeof(t) : 4
t.age : 4
t.age : 7
t.age : 0
```

全文完

本文由 简悦 SimpRead 优化，用以提升阅读体验

使用了 全新的简悦词法分析引擎 beta，点击查看详细说明

