[linux中字符串转换函数](http://blog.csdn.net/tommy_wxie/article/details/7480087)

Linux内核中提供的一些字符串转换函数：

lib/vsprintf.c

**[html]** [view plaincopyprint?](http://blog.csdn.net/tommy_wxie/article/details/7480087)

1. 1. unsigned long long simple\_strtoull(const char \*cp, char \*\*endp, unsigned int base)
2. 2. unsigned long simple\_strtoul(const char \*cp, char \*\*endp, unsigned int base)
3. 3. long simple\_strtol(const char \*cp, char \*\*endp, unsigned int base)
4. 4. long long simple\_strtoll(const char \*cp, char \*\*endp, unsigned int base)
5. 5. int strict\_strtoul(const char \*cp, unsigned int base, unsigned long \*res)
6. 6. int strict\_strtol(const char \*cp, unsigned int base, long \*res)
7. 7. int strict\_strtoull(const char \*cp, unsigned int base, unsigned long long \*res)
8. 8. int strict\_strtoll(const char \*cp, unsigned int base, long long \*res)
9. 9. int sprintf(char \*buf, const char \*fmt, ...)
10. 10. int snprintf(char \*buf, size\_t size, const char \*fmt, ...)
11. 11. int sscanf(const char \*buf, const char \*fmt, ...)

unsigned long long simple\_strtoull(const char \*cp, char \*\*endp, unsigned int base)  
功能：将一个字符串转换成unsigend long long型数据。  
返回：返回转换后数据。

参数：cp指向字符串的开始，endp指向分析的字符串末尾的位置，base为要用的基数（进制数），base为0表示通过cp来自动判断基数，函数自动可识别的基数：‘0x’表示16进制，‘0’表示8进制，其它都认定为10进制。函数可转换成数字的有效字符为：[0，f]。举例：cp = “0x12str”,base = 0，则返回unsigned long long为18，\*endp = “str”。 参数下同。

**[cpp]** [view plaincopyprint?](http://blog.csdn.net/tommy_wxie/article/details/7480087)

1. **static** ssize\_t led\_brightness\_store(**struct** device \*dev,
2. **struct** device\_attribute \*attr, **const** **char** \*buf, **size\_t** size)
3. {
4. **struct** led\_classdev \*led\_cdev = dev\_get\_drvdata(dev);
5. ssize\_t ret = -EINVAL;
6. **char** \*after;
7. unsigned **long** state = simple\_strtoul(buf, &after, 10);
8. **size\_t** count = after - buf;
10. **if** (isspace(\*after))
11. count++;
13. **if** (count == size) {
14. ret = count;
16. **if** (state == LED\_OFF)
17. led\_trigger\_remove(led\_cdev);
18. led\_set\_brightness(led\_cdev, state);
19. }
21. **return** ret;
22. }

unsigned long simple\_strtoul(const char \*cp, char \*\*endp, unsigned int base)  
功能：将一个字符串转换成unsigend long型数据。  
返回：返回转换后数据。  
  
int strict\_strtoul(const char \*cp, unsigned int base, unsigned long \*res)  
功能：将一个字符串转换成unsigend long型。  
返回：转换成功返回0，否则返回负。res指向转换后的unsigned long数据。

说明：该函数对cp指向的字符串严格要求，cp指向的字符串必须为真正的unsigned long形式的字符串。字符串必须以“0x”、“0”、[0,f]开始，中间全部为有效的字符[0,f]，否则返回为负。它会处理字符串最后的“\n”字符。下同

long long simple\_strtoll(const char \*cp, char \*\*endp, unsigned int base)  
功能：将一个字符串转换成sigend long long型。  
返回：返回转换后数据。  
  
  
int strict\_strtol(const char \*cp, unsigned int base, long \*res)  
功能：将一个字符串转换sigend long型。  
返回：转换成功返回0，否则返回负。res指向转换后的signed long数据。  
  
int strict\_strtoull(const char \*cp, unsigned int base, unsigned long long \*res)  
功能：将一个字符串转换unsigend long long型。  
返回：转换成功返回0，否则返回负。res指向转换后的unsigned long long数据。  
  
int strict\_strtoll(const char \*cp, unsigned int base, long long \*res)  
功能：将一个字符串转换sigend long long型。  
返回：转换成功返回0，否则返回负。res指向转换后的signed long long数据。  
  
int sprintf(char \*buf, const char \*fmt, ...)  
功能：格式化输出字符串，类似于printf，只是用字符串buf作为输出对象。  
返回：返回写入buf字符串的字符个数。  
  
int snprintf(char \*buf, size\_t size, const char \*fmt, ...)  
功能：格式化输出字符串，类似于printf，只是用字符串buf作为输出对象。其中size为buf的大小（包括‘\0’字符）。  
返回：返回写入buf字符串的字符个数。  
  
int sscanf(const char \*buf, const char \*fmt, ...)  
功能：格式化输入字符串，类似于scanf，只是用字符串buf作为输入对象。  
返回：返回读取buf字符串的字符个数。  
  
  
lib/kasprintf

**[cpp]** [view plaincopyprint?](http://blog.csdn.net/zhangzhaocap/article/details/7230067)

1. **char** \*kasprintf(gfp\_t gfp, **const** **char** \*fmt, ...)

**[cpp]** [view plaincopyprint?](http://blog.csdn.net/tommy_wxie/article/details/7480087)

1. **char** \*kasprintf(gfp\_t gfp, **const** **char** \*fmt, ...)

char \*kasprintf(gfp\_t gfp, const char \*fmt, ...)  
功能：格式化输出字符串到一段且gfp分配的内存中。  
返回：返回指向该内容的字符串指针。