

信号处理程序中产生输出唯一安全的方法是使用 `write` 函数(见 10.1 节)。特别地,调用 `printf` 或 `sprintf` 是不安全的。为了绕开这个不幸的限制,我们开发一些安全的函数,称为 SIO(安全的 I/O)包,可以用来在信号处理程序中打印简单的消息。

```
#include "csapp.h"
```

```
ssize_t sio_putl(long v);
ssize_t sio_puts(char s[]);
```

返回: 如果成功则为传送的字节数, 如果出错, 则为 -1。

```
void sio_error(char s[]);
```

返回: 空。

`sio_putl` 和 `sio_puts` 函数分别向标准输出传送一个 `long` 类型数和一个字符串。`sio_error` 函数打印一条错误消息并终止。

图 8-34 给出的是 SIO 包的实现, 它使用了 `csapp.c` 中两个私有的可重入函数。第 3 行的 `sio_strlen` 函数返回字符串 `s` 的长度。第 10 行的 `sio_ltoa` 函数基于来自 [61] 的 `itoa` 函数, 把 `v` 转换成它的基 `b` 字符串表示, 保存在 `s` 中。第 17 行的 `_exit` 函数是 `exit` 的一个异步信号安全的变种。

code/src/csapp.c

```
1  ssize_t sio_puts(char s[]) /* Put string */
2  {
3      return write(STDOUT_FILENO, s, sio_strlen(s));
4  }
5
6  ssize_t sio_putl(long v) /* Put long */
7  {
8      char s[128];
9
10     sio_ltoa(v, s, 10); /* Based on K&R itoa() */
11     return sio_puts(s);
12 }
13
14 void sio_error(char s[]) /* Put error message and exit */
15 {
16     sio_puts(s);
17     _exit(1);
18 }
```

code/src/csapp.c

图 8-34 信号处理程序的 SIO(安全 I/O)包

图 8-35 给出了图 8-30 中 SIGINT 处理程序的一个安全的版本。

code/ecf/sigintsafe.c

```
1  #include "csapp.h"
2
3  void sigint_handler(int sig) /* Safe SIGINT handler */
4  {
5      Sio_puts("Caught SIGINT!\n"); /* Safe output */
6      _exit(0); /* Safe exit */
7  }
```

code/ecf/sigintsafe.c

图 8-35 图 8-30 的 SIGINT 处理程序的一个安全版本