### 5.3.4 消息队列（MSG）

回忆前面所述的管道，这种通信机制的一个弊端是：你无法在管道中读取一个“指定”的数据，因为这些数据没有做任何标记，读者进程只能按次序地挨个读取，因此多对进程之间的相互通信，除非使用多条管道分别处理，否则无法使用一条管道来完成。


图 5-30 管道中的无标识数据

消息队列提供一种带有数据标识的特殊管道，使得每一段被写入的数据都变成带标识的消息，读取该段消息的进程只要指定这个标识就可以正确地读取，而不会受到其他消息的干扰，从运行效果来看，一个带标识的消息队列，就像多条并存的管道一样。


图 5-31 消息队列中带标识的数据

消息队列的使用方法一般是：

1，发送者：

A) 获取消息队列的 ID

B) 将数据放入一个附带有标识的特殊的结构体，发送给消息队列。

2，接收者：

A) 获取消息队列的 ID

B) 将指定标识的消息读出。

当发送者和接收者都不再使用消息队列时，及时删除它以释放系统资源。

下面详细解剖消息队列（MSG）的 API。

| 功能 | 获取消息队列的 ID |
|---|---|
| 头文件 | #include <sys/types.h><br>#include <sys/ipc.h><br>#include <sys/msg.h> |
| 原型 | int **msgget**(key_t key, int msgflg); |

| | | | |
|---|---|---|---|
| 参数 | key | 消息队列的键值 | |
| | msgflg | IPC_CREAT | 如果 key 对应的 MSG 不存在，则创建该对象 |
| | | IPC_EXCL | 如果该 key 对应的 MSG 已经存在，则报错 |
| | | mode | MSG 的访问权限（八进制，如 0644） |
| 返回值 | 成功 | 该消息队列的 ID | |
| | 失败 | -1 | |
| 备注 | 如果 key 指定为为 IPC_PRIVATE，则会自动产生一个随机未用的新键值 | | |

图 5-32 函数 msgget()的接口规范

使用该函数需要注意的以下几点：

1，选项 msgflg 是一个位屏蔽字，因此 IPC_CREAT、IPC_EXCL 和权限 mode 可以用位或的方式叠加起来，比如：msgget(key, IPC_CREAT | 0666); 表示如果 key 对应的消息队列不存在就创建，且权限指定为 0666，若已存在则直接获取 ID。

2，权限只有读和写，执行权限是无效的，例如 0777 跟 0666 是等价的。

3，当 key 被指定为 IPC_PRIVATE 时，系统会自动产生一个未用的 key 来对应一个新的消息队列对象。一般用于线程间通信。

| | | | |
|---|---|---|---|
| 功能 | 发送、接收消息 | | |
| 头文件 | #include <sys/types.h><br>#include <sys/ipc.h><br>#include <sys/msg.h> | | |
| 原型 | int **msgsnd**(int msqid, const void *msgp, size_t msgsz, int msgflg);<br>ssize_t **msgrcv**(int msqid, void *msgp,<br>size_t msgsz, long msgtyp, int msgflg); | | |
| 参数 | msqid | 发送、接收消息的消息队列 ID | |
| | msgp | 要发送的数据、要接收的数据的存储区域指针 | |
| | msgsz | 要发送的数据、要接收的数据的大小 | |
| | msgtyp | 这是 msgrcv 独有的参数，代表要接收的消息的标识 | |
| | msgflg | IPC_NOWAIT | 非阻塞读出、写入消息 |
| | | MSG_EXCEPT | 读取标识不等于 msgtyp 的第一个消息 |
| | | MSG_NOERROR | 消息尺寸比 msgsz 大时，截断消息而不报错 |
| 返回值 | 成功 | msgsnd( ) | 0 |
| | | msgrcv( ) | 真正读取的字节数 |
| | 失败 | -1 | |
| 备注 | 无 | | |

图 5-33 消息队列收发消息的函数接口规范

使用这两个收、发消息函数需要注意以下几点：

1，发送消息时，消息必须被组织成以下形式：

```
struct msgbuf
{
    long mtype;     // 消息的标识
    char mtext[1]; // 消息的正文
};
```

也就是说：发送出去的消息必须以一个 long 型数据打头，作为该消息的标识，后面的数据则没有要求。

2，消息的标识可以是任意长整型数值，但不能是 0L。

3，参数 msgsz 是消息中正文的大小，不包含消息的标识。