int PASCAL FAR ioctl[socket](http://baike.baidu.com/view/13870.htm)( [SOCKET](http://baike.baidu.com/view/13870.htm) s, long [cmd](http://baike.baidu.com/view/65856.htm), u\_long FAR\* argp);

关。支持下列命令：

**FIONBIO：**

允许或禁止[套接口](http://baike.baidu.com/view/567586.htm)s的非阻塞模式。argp指向一个无符号[长整型](http://baike.baidu.com/view/3522307.htm)，如允许非阻塞模式则非零，如禁止非阻塞模式则为零。当创建一个[套接口](http://baike.baidu.com/view/567586.htm)时，它就处于阻塞模式（也就是说非阻塞模式被禁止）。这与BSD[套接口](http://baike.baidu.com/view/567586.htm)是一致的。WSAAsynSelect()函数将套接口自动设置为非阻塞模式。如果已对一个[套接口](http://baike.baidu.com/view/567586.htm)进行了WSAAsynSelect() 操作，则任何用ioctl[socket](http://baike.baidu.com/view/13870.htm)()来把套接口]重新设置成阻塞模式的试图将以WSAEINVAL失败。为了把[套接口](http://baike.baidu.com/view/567586.htm)重新设置成阻塞模式，[应用程序](http://baike.baidu.com/view/330120.htm)必须首先用WSAAsynSelect()调用（IEvent参数置为0）来禁止WSAAsynSelect()。

**FIONREAD：**

确定[套接口](http://baike.baidu.com/view/567586.htm)s自动读入的数据量。argp指向一个无符号[长整型](http://baike.baidu.com/view/3522307.htm)，其中存有ioctl[socket](http://baike.baidu.com/view/13870.htm)()的返回值。如果s是SOCKET\_STREAM类型，则FIONREAD返回在一次[recv()](http://baike.baidu.com/view/569210.htm)中所接收的所有数据量。这通常与[套接口](http://baike.baidu.com/view/567586.htm)中排队的数据总量相同。如果S是SOCK\_DGRAM 型，则FIONREAD返回[套接口](http://baike.baidu.com/view/567586.htm)上排队的第一个数据报大小。

**SIOCATMARK：**

确实是否所有的[带外数据](http://baike.baidu.com/view/567593.htm)都已被读入。这个命令仅适用于SOCK\_STREAM类型的[套接口](http://baike.baidu.com/view/567586.htm)，且该套接口已被设置为可以在线接收[带外数据](http://baike.baidu.com/view/567593.htm)（SO\_OOBINLINE）。如无[带外数据](http://baike.baidu.com/view/567593.htm)等待读入，则该操作返回TRUE真。否则的话返回FALSE假，下一个[recv()](http://baike.baidu.com/view/569210.htm)或[recvfrom()](http://baike.baidu.com/view/569211.htm)操作将检索“标记”前一些或所有数据。[应用程序](http://baike.baidu.com/view/330120.htm)可用SIOCATMARK操作来确定是否有数据剩下。如果在“紧急”（带外）数据[前有常规数据，则按序接收这些数据（请注意，[recv()](http://baike.baidu.com/view/569210.htm)和[recvfrom()](http://baike.baidu.com/view/569211.htm)操作不会在一次调用中混淆常规数据与带外数]据）。argp指向一个BOOL型数，ioctl[socket](http://baike.baidu.com/view/13870.htm)()在其中存入返回值。

S

[Remarks](javascript:void(0))

This function can be used on any socket in any state. It is used to set or retrieve operating parameters associated with the socket, independent of the protocol and communications subsystem. The supported commands to use in the cmd parameter and their semantics are as follows.

FIONBIO

Use FIONBIO with a nonzero argp parameter to enable the nonblocking mode of socket s. The argp parameter is zero if nonblocking is to be disabled. The argp parameter points to an unsigned long value. When a socket is created, it operates in blocking mode by default (nonblocking mode is disabled). This is consistent with BSD sockets.

The [WSAEventSelect](http://msdn.microsoft.com/en-us/library/aa922328.aspx) function automatically sets a socket to nonblocking mode. If **WSAEventSelect** has been issued on a socket, then any attempt to use **ioctlsocket** to set the socket back to blocking mode will fail with WSAEINVAL.

To set the socket back to blocking mode, an application must first disable **WSAEventSelect** by calling **WSAEventSelect** with the lNetworkEvents parameter equal to zero.

FIONREAD

Use FIONREAD to determine the amount of data pending in the network's input buffer that can be read from socket s. The argp parameter points to an unsigned long value in which **ioctlsocket** stores the result. If s is stream-oriented (for example, type **SOCK\_STREAM**), FIONREAD returns the amount of data that can be read in a single call to the [recv](http://msdn.microsoft.com/en-us/library/aa922642.aspx) function; this might not be the same as the total amount of data queued on the socket. If s is message-oriented (for example, type **SOCK\_DGRAM**), FIONREAD returns the size of the first datagram (message) queued on the socket.

SIOCATMARK

SIOCATMARK is not supported and returns WSAEINVAL.

nRet = ioctlsocket(sockClient, FIONREAD, &ul);//查查看緩衝區多少字節數據可以被recv

nRet = recv(sockClient, szBuffer, MAX\_PATH, 0);

int BlockSocket(SOCKET s)

{

u\_long l = 0L;

return ioctlsocket(s, FIONBIO, &l);

}

int UnBlockSocket(SOCKET s)

{

u\_long l = 1L;

return ioctlsocket(s, FIONBIO, &l);

}