SerShk structure

#include <Serial.h>

typedef struct **SerShk** {		Size	Offset	Description
char	fXOn;	1	0	XOn/XOff output flow control
char	fCTS;	1	1	Clear to Send hardware handshake
char	xOn;	1	2	XOn character
char	xOff;	1	3	XOff character
char	errs;	1	4	Errors resulting in abort
char	evts;	1	5	Status changes that cause events
char	fInX;	1	6	XOn/XOff input flow control
char	fDTR;	1	7	Data Terminal Ready input flow control
} **SerShk**;		8		

Notes: This structure represents the contents of the flags parameter in the SerHShake procedure and addresses the input and output drivers as identified by the call's refNum field. It is used to set handshaking and control parameters as follows:

Output flow control is enabled when fXOn is set, disabled when cleared to zero. The same holds true for FInX (as regard input flow control. The xOn and xOff fields specify XOn and XOff characters for flow control. Setting the fCTS enables the Clear to Send (CTS) hardware handshake.

The errs field shows the values:

parityErr	(16)	set if a parity error has occurred
hwOverrunErr	(32)	set in the event of a hardware overrun error
framingErr	(64)	set in the event of a framing error

Any one of these errors will cause input requests to abort.

The evts field shows whether or not changes in Clear to Send or Break status will make the **Serial Driver** post a device driver event as follows:

ctsEvent	(32)	set if Clear to Send change will post an event
breakEvent	(128)	set if Break status change will post an event

Your applications can use these constants to set or test the value of the evts field but we're warned against it because posting events disables interrupts for an inordinately long time.