NAME
sctp_connectx - connect an SCTP socket with multiple destination addresses

LIBRARY
Standard C Library (libc, -lc)

SYNOPSIS
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/sctp.h>

int
sctp_connectx(*int sd*, *struct sockaddr *addrs*, *int addrcnt*, *sctp_assoc_t *id*);

DESCRIPTION
The **sctp_connectx**() call attempts to initiate an association to a peer SCTP endpoint. The call operates similarly to **connect**() but it also provides the ability to specify multiple destination addresses for the peer. This allows a fault tolerant method of initiating an association. When one of the peers addresses is unreachable, the subsequent listed addresses will also be used to set up the association with the peer.

The user also needs to consider that any address listed in an **sctp_connectx**() call is also considered "confirmed". A confirmed address is one in which the SCTP transport will trust is a part of the association and it will not send a confirmation heartbeat to it with a random nonce.

If the peer SCTP stack does not list one or more of the provided addresses in its response message then the extra addresses sent in the **sctp_connectx**() call will be silently discarded from the association. On successful completion the provided *id* will be filled in with the association identification of the newly forming association.

RETURN VALUES
The call returns 0 on success and -1 upon failure.

ERRORS
The **sctp_connectx**() function can return the following errors:

[EINVAL]   An address listed has an invalid family or no addresses were provided.

[E2BIG]   The size of the address list exceeds the amount of data provided.

[EBADF]   The argument *s* is not a valid descriptor.

[ENOTSOCK]   The argument *s* is not a socket.

SEE ALSO
  connect(2), sctp(4)