─────────────────── MODULE $SCTP$ ───────────────────

LOCAL INSTANCE $Naturals$

LOCAL INSTANCE $Sequences$

LOCAL INSTANCE $FiniteSets$

LOCAL INSTANCE TLC

───

CONSTANT Nil

LOCAL $Min(s) \triangleq$ CHOOSE $x \in s : \forall\, y \in s : x \geq y$

LOCAL $Max(s) \triangleq$ CHOOSE $x \in s : \forall\, y \in s : x \leq y$

VARIABLE $conns$

$vars \triangleq \langle conns \rangle$

─────────────────── MODULE $Client$ ───────────────────

CONSTANT ID

$Connect(dst) \triangleq$
    $\wedge\ dst \in$ DOMAIN $conns$
    $\wedge$ LET $maxId \triangleq Max(\{conns[dst][i].connId : i \in conns[dst]\})$
           $connId \triangleq Min(\{i \in 1\,..\,(maxId+1) : i \notin$ DOMAIN $conns[dst]\})$
           $conn \triangleq [id\ \mapsto connId,$
                   $src \mapsto ID,$
                   $dst \mapsto dst,$
                   $req \mapsto \langle\rangle,$
                   $res \mapsto \langle\rangle]$
       IN $conns' = [conns$ EXCEPT $![dst] = conns[dst] @@ (connId :> conn)]$

$Disconnect(conn) \triangleq$
    $conns' = [conns$ EXCEPT $![conn.dst] = [x \in$ DOMAIN $conns[conn.dst] \setminus \{conn.id\} \mapsto conns[conn.dst][x]]]$

$Send(conn,\ msg) \triangleq$
    $conns' = [conns$ EXCEPT $![conn.dst] = [$
           $conns[conn.dst]$ EXCEPT $![conn.id] = [$
              $conns[conn.dst][conn.id]$ EXCEPT $!.req = Append(conns[conn.dst][conn.id].req,\ msg)]]]$

$Receive(conn) \triangleq$
    $conns' = [conns$ EXCEPT $![conn.dst] = [$
           $conns[conn.dst]$ EXCEPT $![conn.id] = [$
              $conns[conn.dst][conn.id]$ EXCEPT $!.res = SubSeq(conns[conn.dst][conn.id].res,\ 2,\ Len(co$

$Reply(conn, msg) \triangleq$
$\quad conns' = [conns'$ EXCEPT $![conn.dst] = [$
$\qquad\qquad conns'[conn.dst]$ EXCEPT $![conn.id] = [$
$\qquad\qquad\qquad conns'[conn.dst][conn.id]$ EXCEPT $!.req = Append(conns'[conn.dst][conn.id].req, msg)]]]$

$Handle(conn, handler(\_, \_)) \triangleq Len(conn.res) > 0 \land handler(conn, conn.res[1])$

$Connections \triangleq \{conn \in$ UNION $\{\{conns[s][c] : c \in$ DOMAIN $s\} : s \in conns\} : conn.src = ID\}$

$Client(ID) \triangleq$ INSTANCE $Client$

---------------- MODULE $Server$ ----------------

CONSTANT ID

$Start \triangleq$
$\quad \land ID \notin$ DOMAIN $conns$
$\quad \land conns' = conns @@ (ID :> [connId \in \{\} \mapsto [connId \mapsto connId]])$

$Stop \triangleq$
$\quad \land ID \in$ DOMAIN $conns$
$\quad \land conns' = [c \in \{c \in$ DOMAIN $conns : c \neq ID\} \mapsto conns[c]]$

$Send(conn, msg) \triangleq$
$\quad \land Assert(conn.dst = ID,$ "Send on invalid connection"$)$
$\quad \land conns' = [conns$ EXCEPT $![conn.dst] = [$
$\qquad\qquad conns[conn.dst]$ EXCEPT $![conn.id] = [$
$\qquad\qquad\qquad conns[conn.dst][conn.id]$ EXCEPT $!.res = Append(conns[conn.dst][conn.id].res, msg)]]$

$Receive(conn) \triangleq$
$\quad \land Assert(conn.dst = ID,$ "Receive on invalid connection"$)$
$\quad \land conns' = [conns$ EXCEPT $![conn.dst] = [$
$\qquad\qquad conns[conn.dst]$ EXCEPT $![conn.id] = [$
$\qquad\qquad\qquad conns[conn.dst][conn.id]$ EXCEPT $!.res = SubSeq(conns[conn.dst][conn.id].req, 2, Len$

$Reply(conn, msg) \triangleq$
$\quad \land Assert(conn.dst = ID,$ "Reply on invalid connection"$)$
$\quad \land conns' = [conns'$ EXCEPT $![conn.dst] = [$
$\qquad\qquad conns'[conn.dst]$ EXCEPT $![conn.id] = [$
$\qquad\qquad\qquad conns'[conn.dst][conn.id]$ EXCEPT $!.req = Append(conns'[conn.dst][conn.id].res, msg)]]]$

$Handle(conn, handler(\_, \_)) \triangleq Len(conn.req) > 0 \land handler(conn, conn.req[1])$

$Connections \triangleq \{conn \in$ UNION $\{\{conns[s][c] : c \in$ DOMAIN $s\} : s \in conns\} : conn.dst = ID\}$

$Server(ID) \triangleq$ INSTANCE $Server$

2

$Init \;\triangleq$
 $\wedge\; conns = [id \in \{\} \mapsto [$
       $connId \in \{\} \mapsto [connId \mapsto connId,$
             $src \quad\;\; \mapsto Nil,$
             $dst \quad\;\; \mapsto Nil,$
             $req \quad\;\; \mapsto \langle\rangle,$
             $res \quad\;\; \mapsto \langle\rangle]]]$

$Next \;\triangleq$
 $\vee\; \text{UNCHANGED} \;\langle conns\rangle$

\ * Modification History
\ * Last modified *Tue Sep* 21 08:58:31 *PDT* 2021 by *jordanhalterman*
\ * Created *Mon Sep* 13 12:21:16 *PDT* 2021 by *jordanhalterman*