MODULE *htlc*

Specifications for the *HTLC* sending and forwarding. The protocol is composed of actions like initiate, update, expire. These actions specify how the state of each node and the balance on each channel is allowed to change in response to handling *HTLC* messages

EXTENDS *Integers*,
TLC

CONSTANTS *Node*, *Channel*, *InitialBalance*

Channels are unidirectional in the spec. This helps us track states and balances for the purposes of the specifications.

VARIABLES *htcl_states*,
channel_balances

$vars \triangleq \langle htcl\_states,\ channel\_balances \rangle$

$update\_states \triangleq \{$"ready",
"pending",
"in_latest_commit_tx",
"prev_commit_tx_revoked"$\}$

Initialise with an initial balance and ready state

$Init \triangleq$
$\land\ channel\_balances = [\langle m,\ n \rangle \in Channel \mapsto \text{CHOOSE } b \in InitialBalance : \text{TRUE}]$
$\land\ htcl\_states = [\langle m,\ n \rangle \in Channel \mapsto$ "ready"$]$

$TypeInvariant \triangleq$
$\land\ Channel \in Node \times Node$ channels are between nodes
$\land\ channel\_balances \in [Node \times Node \to InitialBalance]$ channel balance
$\land\ htcl\_states \in [Node \times Node \to update\_states]$ channels *htlc* state

---

When invoked on channel $\langle a,\ b \rangle$. The commit transaction of b is affected.

$update\_add\_htlc(m,\ n,\ amount) \triangleq$
$\land\ htcl\_states[\langle m,\ n \rangle] =$ "ready" Commit tx state should be ready
$\land\ channel\_balances[\langle m,\ n \rangle] > 0$ Forward only if there is some balance
$\land\ htcl\_states' = [htcl\_states \text{ EXCEPT } ![\langle m,\ n \rangle] =$ "pending"$]$ Change state to pending
$\land$ UNCHANGED *channel_balances*

$Next \triangleq$
$\lor\ \exists \langle m,\ n \rangle \in Channel,\ a \in InitialBalance :$

$\land\ update\_add\_htlc(m,\ n,\ a)$

$Spec\ \triangleq$
$\quad \land\ Init$
$\quad \land\ \Box[Next]_{\langle vars \rangle}$