MODULE *htlc*

Specifications for the *HTLC* sending and forwarding. The protocol is composed of a number of actions like initiate, update, expire. These actions collectively specify how the state of each node and the balance on each channel can change.

EXTENDS *Integers*

Channel is considered directional in this specification. So $\langle a, b \rangle$ is a channel and so is $\langle b, a \rangle$

CONSTANTS *Channel*, *InitialBalance*

VARIABLES *balance*, *commit_txs*

$vars \triangleq \langle balance \rangle$

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

$Init \triangleq$
$\quad \land \forall\, c \in Channel : balance[c] = \text{CHOOSE } b : b \in InitialBalance$ Initialise with any given initial balance

$TypeInvariant \triangleq$
$\quad \land\ balance \in [Channel \to Int]$
There are no constraints in the protocol on the state of the counterparties states. So all combinations are allowed
$\quad \land\ commit\_txs \in [Channel \to node\_states \times node\_states]$

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

$update\_add\_htlc(channel, amount) \triangleq$
$\quad \land\ commit\_txs[channel] =$ "ready" *commit_tx* on b's side is in ready state
$\quad \land\ commit\_txs' = [commit\_txs \text{ EXCEPT } ![channel] =$ "pending"$]$ *commit_tx* on b's moves to pending