
MODULE *BitcoinTransactionsSpec*

This *Spec* is used to run a model against the *BitcoinTransactions* module.

By moving the model and the runner here, we allow other modules like *LNContracts* to freely use *BitcoinTransactions* and run their own models.

EXTENDS *BitcoinTransactions*

$vars \triangleq \langle chain_height, transactions, mempool, published \rangle$

$Init \triangleq$
 $\wedge transactions = [id \in TXID \mapsto [inputs \mapsto \langle \rangle, outputs \mapsto \langle \rangle]]$
 $\wedge chain_height = 0$
 $\wedge mempool = \{\}$
 $\wedge published = [id \in TXID \mapsto NoSpendHeight]$

$TypeOK \triangleq$
 $\wedge transactions \in [TXID \rightarrow [inputs : Seq(Input), outputs : Seq(Output)]]$
 $\wedge mempool \in SUBSET\ TXID$
 $\wedge published \in [TXID \rightarrow Int]$

$Next \triangleq$
 $\vee \exists k \in Keys, id \in TXID, a \in AMOUNT :$
 $\quad \vee AddP2WKHCoinbaseToMempool(id, \langle k \rangle, a)$
 $\vee \exists keys \in Keys \times Keys, id \in TXID, amount \in AMOUNT :$
 $\quad \vee AddMultisigCoinbaseToMempool(id, keys, amount)$
 $\vee \exists id \in TXID, a \in AMOUNT, input_type \in OutputTypes, output_type \in OutputTypes :$
 $\quad AddSpendTxToMempool(id, a, input_type, output_type)$
 $\vee \exists id \in TXID : ConfirmMempoolTx(id)$

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