```
MODULE IscpBatchTimestamp

  asd

                      ∀i : a_i = b

  Can we determine if proposal is invalid with regards to the timestamp? That is done after the
  ACS.

EXTENDS Naturals, FiniteSets, TLAPS
CONSTANT Nodes          A set of node identifiers.
CONSTANT Byzantine      A set of byzantine node identifiers.
CONSTANT Time           A set of timestamps, represented as natural numbers to have ≤ .
ASSUME ConstantAssms ≜ Byzantine ⊆ Nodes ∧ Time ⊆ Nat
Requests ≜ Time   Assume requests are identified by timestamps of their TX only.

VARIABLE proposed   Was the proposal made?
VARIABLE npRq        Node proposal: A set of requests.
VARIABLE npTS       Node proposal: Timestamp.
vars ≜ ⟨proposed, npRq, npTS⟩

F ≜ Cardinality(Byzantine)
N ≜ Cardinality(Nodes)
ASSUME ByzantineAssm ≜ N ≥ 3 ∗ F + 1

F1Quorums ≜ {q ∈ SUBSET Nodes : Cardinality(q) = F + 1}
NFQuorums ≜ {q ∈ SUBSET Nodes : Cardinality(q) = N − F}

BatchRq(rq) ≜ ∃ q ∈ F1Quorums : ∀ n ∈ q : rq ∈ npRq[n]
BatchRqs    ≜ {rq ∈ Requests : BatchRq(rq)}

SubsetTS(s) ≜ {npTS[n] : n ∈ s}
BatchTS(ts) ≜ ∃ q ∈ NFQuorums :
                   ∧ ts ∈ SubsetTS(q)
                   ∧ ∀ x ∈ SubsetTS(q) : ts ≥ x
                   ∧ ∀ x ∈ SubsetTS(Nodes \ q) : ts ≤ x

ProposalValid(n) ≜ ∀ rq ∈ npRq[n] : rq ≤ npTS[n]
Propose ≜ ¬proposed ∧ proposed′ = TRUE
   ∧ npRq′ ∈ [Nodes → (SUBSET Requests) \ {{}}]      Some node non-empty proposals.
   ∧ npTS′ ∈ [Nodes → Time]                          Some timestamps.
   ∧ ∀ n ∈ (Nodes \ Byzantine) : ProposalValid(n)′   Fair node proposals are valid.
──────────────────────────────────────────────────────────────────────────────
Init ≜ proposed = FALSE ∧ npRq = {} ∧ npTS = {}   Dummy values, on init.
Spec ≜ Init ∧ □[Propose]_vars                     For model checking in TLC.
Invariant ≜ proposed ⇒ ∀ ts ∈ Time, rq ∈ BatchRqs : BatchTS(ts) ⇒ rq ≤ ts

THEOREM Spec ⇒ □Invariant
  PROOF OMITTED   Checked with TLC.
THEOREM Byzantine = {} ∧ Spec ⇒ □Invariant
```

```
⟨1⟩1. Byzantine      = {} ∧ Init  ⇒ Invariant
⟨1⟩2. Invariant ∧ Byzantine = {} ∧ Propose ⇒ Invariant′
⟨1⟩q. QED BY ⟨1⟩1, ⟨1⟩2, PTL, ConstantAssms DEF Spec
```

Counter-example with $Nodes = 101 \,..\, 104$, $Byzantine = \{104\}$, $Time = 1 \,..\, 3$:
$PropposedRq$: $(101 :> \{1\} \,@@\, 102 :> \{1\} \,@@\, 103 :> \{2\} \,@@\, 104 :> \{2\})$,
$PropposedTS$: $(101 :> 1 \,@@\, 102 :> 1 \,@@\, 103 :> 2 \,@@\, 104 :> 1\ )$,
$BatchRq$: $\{1, 2\}$,
$BatchTS$: 1