```
┌──────────────────────── MODULE RIC ────────────────────────┐
LOCAL INSTANCE Naturals

LOCAL INSTANCE Sequences

LOCAL INSTANCE FiniteSets

LOCAL INSTANCE TLC

├────────────────────────────────────────────────────────────┤

 An empty value
CONSTANT Nil

 Node states
CONSTANT Stopped, Started

 A set of RIC node identifiers
CONSTANT RICNode

ASSUME  ∧ IsFiniteSet(RICNode)
        ∧ ∀ n ∈ RICNode : n ∈ STRING

 A mapping of node states
VARIABLE state

 Network state
VARIABLE network

 A global store of E2 node states
VARIABLE nodes

 A set of E2 node management connections
VARIABLE mgmtConn

 A transaction ID counter
VARIABLE txId

 A set of E2 node data connections
VARIABLE dataConn

 A request ID counter
VARIABLE reqId

vars ≜ ⟨state, network, nodes, mgmtConn, txId, dataConn, reqId⟩

LOCAL E2AP ≜ INSTANCE E2AP WITH conns ← network

├────────────────────────────────────────────────────────────┤
```

```
$StartNode(node) \triangleq$
    $\land state[node] = Stopped$
    $\land E2AP!Server(node)!Start$
    $\land state' = [nodes \text{ EXCEPT } ![node] = Started]$
    $\land \text{UNCHANGED } \langle nodes \rangle$

$StopNode(node) \triangleq$
    $\land state[node] = Started$
    $\land E2AP!Server(node)!Stop$
    $\land state' = [nodes \text{ EXCEPT } ![node] = Stopped]$
    $\land \text{UNCHANGED } \langle network, nodes \rangle$

--------------------------------------------------------------------------------

$HandleE2SetupRequest(node, conn, req) \triangleq$
    $\land E2AP!Server(node)!Receive!E2SetupRequest(conn, req)$
    $\land \lor \land conn.id \notin \text{DOMAIN } mgmtConn$
          $\land mgmtConn' = mgmtConn @@ (conn.id :> conn)$
       $\lor \land conn.id \in \text{DOMAIN } mgmtConn$
          $\land \text{UNCHANGED } \langle mgmtConn \rangle$
    $\land \lor \land req.e2NodeId \notin \text{DOMAIN } nodes$
          $\land nodes' = nodes @@ (req.e2NodeId :> [e2NodeId \mapsto req.e2NodeId,$
                                                    $sms \mapsto req.sms])$
       $\lor \land req.e2NodeId \in \text{DOMAIN } nodes$
          $\land nodes' = [nodes \text{ EXCEPT } ![req.e2NodeId] = [$
                        $nodes[req.e2NodeId] \text{ EXCEPT } !.sms = req.sms]]$
    $\land E2AP!Server(node)!Reply!E2SetupResponse(conn, [txId \mapsto req.txId])$
    $\land \text{UNCHANGED } \langle state, txId, reqId \rangle$

$HandleRICControlResponse(node, conn, res) \triangleq$
    $\land E2AP!Server(node)!Receive!RICControlResponse(conn, res)$
    $\land \text{UNCHANGED } \langle state \rangle$

$HandleRICSubscriptionResponse(node, conn, res) \triangleq$
    $\land E2AP!Server(node)!Receive!RICSubscriptionResponse(conn, res)$
    $\land \text{UNCHANGED } \langle state \rangle$

$HandleRICSubscriptionDeleteResponse(node, conn, res) \triangleq$
    $\land E2AP!Server(node)!Receive!RICSubscriptionDeleteResponse(conn, res)$
    $\land \text{UNCHANGED } \langle state \rangle$

$HandleRICIndication(node, conn, res) \triangleq$
    $\land E2AP!Server(node)!Receive!RICIndication(conn, res)$
    $\land \text{UNCHANGED } \langle state \rangle$

$E2ConnectionUpdate(node) \triangleq$
    $\land node \in \text{DOMAIN } mgmtConn$
    ∧ E2AP!Server(node)!Send!E2ConnectionUpdate(mgmtConn[node], [txId ↦ txId[node] + 1])
    ∧ txId′ = [txId EXCEPT ![node] = txId[node] + 1]
    ∧ UNCHANGED ⟨state⟩

HandleE2NodeConfigurationUpdate(node, conn, req) ≜
    ∧ E2AP!Server(node)!Receive!E2NodeConfigurationUpdate(conn, req)
    ∧ E2AP!Server(node)!Reply!E2NodeConfigurationUpdateAcknowledge(conn, [txId ↦ req.txId])
    ∧ UNCHANGED ⟨state⟩

Init ≜
    ∧ E2AP!Init
    ∧ state  = [n ∈ RICNode ↦ Stopped]
    ∧ nodes = [n ∈ {} ↦ [id ↦ Nil]]
    ∧ mgmtConn = [n ∈ {} ↦ Nil]
    ∧ txId = [n ∈ RICNode ↦ 0]
    ∧ dataConn = [n ∈ {} ↦ Nil]
    ∧ reqId = [n ∈ RICNode ↦ 0]

Next ≜
    ∨ ∃ node ∈ RICNode : StartNode(node)
    ∨ ∃ node ∈ RICNode : StopNode(node)
    ∨ ∃ node ∈ RICNode : E2ConnectionUpdate(node)
    ∨ ∃ node ∈ RICNode :
        ∃ conn ∈ E2AP!Server(node)!Connections :
          ∨ E2AP!Server(node)!Handle!E2SetupRequest(conn, LAMBDA c, m : HandleE2SetupRequest(node,
          ∨ E2AP!Server(node)!Handle!RICControlResponse(conn, LAMBDA c, m : HandleRICControlRespo
          ∨ E2AP!Server(node)!Handle!RICSubscriptionResponse(conn, LAMBDA c, m : HandleRICSubscripti
          ∨ E2AP!Server(node)!Handle!RICSubscriptionDeleteResponse(conn, LAMBDA c, m : HandleRICSu
          ∨ E2AP!Server(node)!Handle!RICIndication(conn, LAMBDA c, m : HandleRICIndication(node, co
          ∨ E2AP!Server(node)!Handle!E2NodeConfigurationUpdate(conn, LAMBDA c, m : HandleE2NodeCo

\* Modification History
\* Last modified Tue Sep 21 10:06:48 PDT 2021 by jordanhalterman
\* Created Tue Sep 21 02:14:49 PDT 2021 by jordanhalterman
```