```
──────────────────── MODULE SB ────────────────────
EXTENDS Naturals, Sequences, TLC, Messages

 Message type constants
CONSTANT
    E2Setup,
    E2SetupResponse,
    E2SetupFailure
CONSTANT
    ResetRequest,
    ResetResponse
CONSTANT
    RICSubscriptionRequest,
    RICSubscriptionResponse,
    RICSubscriptionFailure
CONSTANT
    RICSubscriptionDeleteRequest,
    RICSubscriptionDeleteResponse,
    RICSubscriptionDeleteFailure
CONSTANT
    RICControlRequest,
    RICControlResponse,
    RICControlFailure,
    RICServiceUpdate
CONSTANT
    E2ConnectionUpdate,
    E2ConnectionUpdateAcknowledge,
    E2ConnectionUpdateFailure
CONSTANT
    E2NodeConfigurationUpdate,
    E2NodeConfigurationUpdateAcknowledge,
    E2NodeConfigurationUpdateFailure

 Failure cause constants
CONSTANT
    MiscFailureUnspecified,
    MiscFailureControlProcessingOverload,
    MiscFailureHardwareFailure,
    MiscFailureOMIntervention
CONSTANT
    ProtocolFailureUnspecified,
    ProtocolFailureTransferSyntaxError,
    ProtocolFailureAbstractSyntaxErrorReject,
    ProtocolFailureAbstractSyntaxErrorIgnoreAndNotify,
    ProtocolFailureMessageNotCompatibleWithReceiverState,
```

ProtocolFailureSemanticError,
ProtocolFailureAbstractSyntaxErrorFalselyConstructedMessage

CONSTANT
RICFailureUnspecified,
RICFailureRANFunctionIDInvalid,
RICFailureActionNotSupported,
RICFailureExcessiveActions,
RICFailureDuplicateAction,
RICFailureDuplicateEvent,
RICFailureFunctionResourceLimit,
RICFailureRequestIDUnknown,
RICFailureInconsistentActionSubsequentActionSequence,
RICFailureControlMessageInvalid,
RICFailureCallProcessIDInvalid

CONSTANT
RICServiceFailureUnspecified,
RICServiceFailureFunctionNotRequired,
RICServiceFailureExcessiveFunctions,
RICServiceFailureRICResourceLimit

CONSTANT
TransportFailureUnspecified,
TransportFailureTransportResourceUnavailable

VARIABLE *sbConn*

VARIABLE *sbConnId*

$\mathit{InitSBVars} \triangleq$
$\quad \land\ \mathit{sbConn} = [c \in \{\} \mapsto [\mathit{e2node} \mapsto \mathit{Nil},\ \mathit{ricnode} \mapsto \mathit{Nil},\ \mathit{messages} \mapsto \langle\rangle]]$

$\mathit{SBSend}(c,\ m)\ \triangleq\ \mathit{sbConn}' = [\mathit{sbConn}\ \text{EXCEPT}\ ![c] = [\mathit{sbConn}[c]\ \text{EXCEPT}\ !.\mathit{messages} = \mathit{Append}(\mathit{sbConn}[c].m$

$\mathit{SBReceive}(c)\ \triangleq\ \mathit{sbConn}' = [\mathit{sbConn}\ \text{EXCEPT}\ ![c] = [\mathit{sbConn}[c]\ \text{EXCEPT}\ !.\mathit{messages} = \mathit{SubSeq}(\mathit{sbConn}[c].\mathit{me}$

$\mathit{SBReply}(c,\ m)\ \triangleq\ \mathit{sbConn}' = [\mathit{sbConn}\ \text{EXCEPT}\ ![c] = [\mathit{sbConn}[c]\ \text{EXCEPT}\ !.\mathit{messages} = \mathit{Append}(\mathit{SubSeq}(\mathit{sbCo}$

$\mathit{SBConnect}(\mathit{e2node},\ \mathit{ricnode})\ \triangleq$
$\quad \land\ \mathit{sbConnId}' = \mathit{sbConnId} + 1$
$\quad \land\ \mathit{sbConn}' = \mathit{sbConn}\ @@\ (\mathit{sbConnId}' :> [\mathit{e2node} \mapsto \mathit{e2node},\ \mathit{ricnode} \mapsto \mathit{ricnode},\ \mathit{messages} \mapsto \langle\rangle])$

$\mathit{SBDisconnect}(\mathit{id})\ \triangleq$
$\quad \land\ \mathit{sbConn}' = [c \in \{v \in \text{DOMAIN}\ \mathit{sbConn} : v \neq \mathit{id}\} \mapsto \mathit{sbConn}[c]]$

$$
\begin{array}{l}
\mathit{SBNext} \triangleq \\
\quad \lor \;\; \exists\, c \in \textsc{domain}\ \mathit{sbConn} : \mathit{SBDisconnect}(c)
\end{array}
$$

* Modification History

* Last modified *Tue Aug* 03 18:57:12 *PDT* 2021 by *jordanhalterman*

* Created *Mon Jul* 26 10:01:02 *PDT* 2021 by *jordanhalterman*