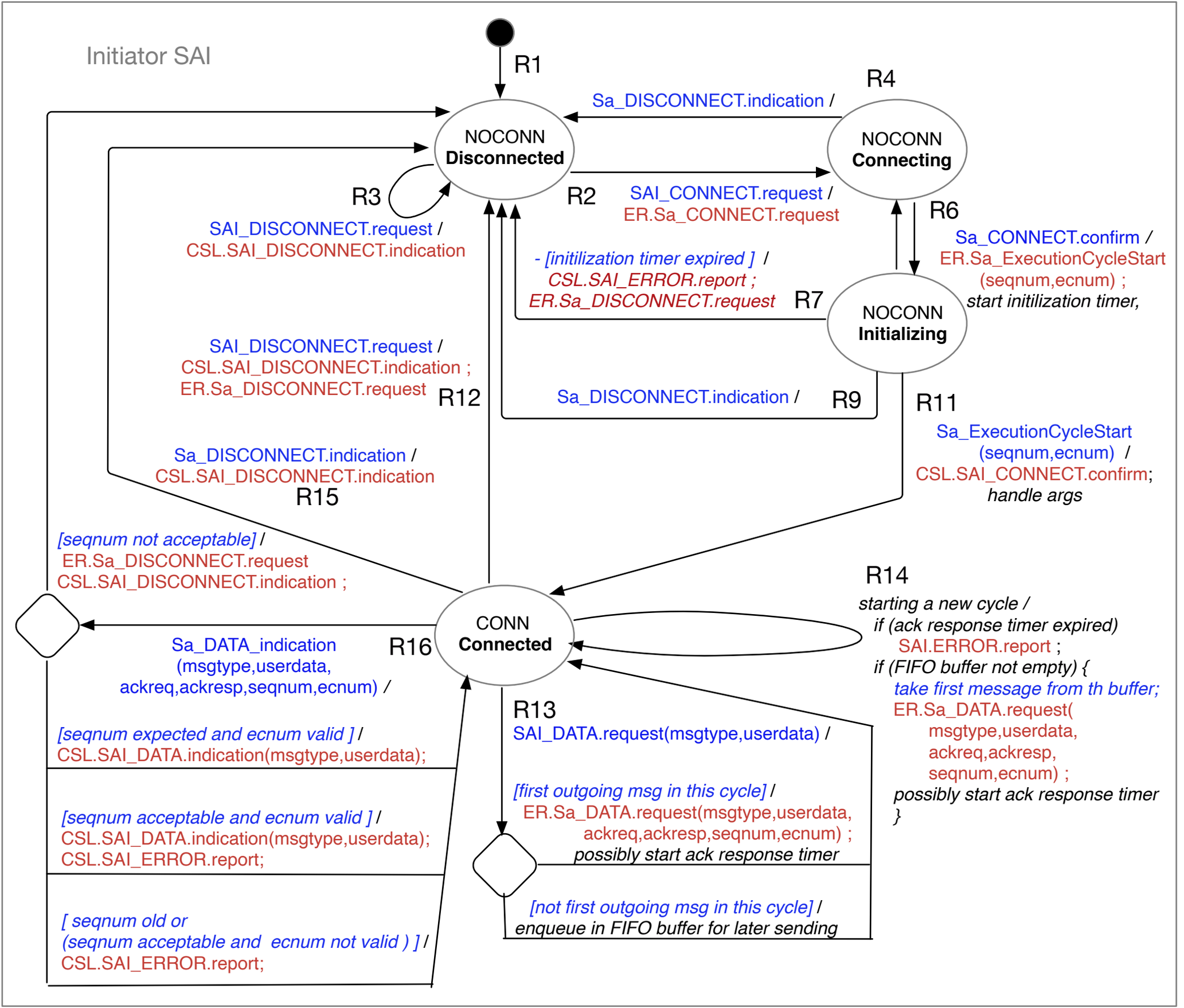
Initiator SAI
R1
NOCONN
Disconnected
R4
Sa_DISCONNECT.indication /
NOCONN
Connecting
R3
SAI_DISCONNECT.request /
CSL.SAI_DISCONNECT.indication
R2
SAI_CONNECT.request /
ER.Sa_CONNECT.request
R6
Sa_CONNECT.confirm /
ER.Sa_ExecutionCycleStart
(seqnum,ecnum) ;
start initilization timer,
- [initilization timer expired] /
CSL.SAI_ERROR.report ;
ER.Sa_DISCONNECT.request
R7
NOCONN
Initializing
SAI_DISCONNECT.request /
CSL.SAI_DISCONNECT.indication ;
ER.Sa_DISCONNECT.request
R12
Sa_DISCONNECT.indication /
R9
R11
Sa_ExecutionCycleStart
(seqnum,ecnum) /
CSL.SAI_CONNECT.confirm;
handle args
Sa_DISCONNECT.indication /
CSL.SAI_DISCONNECT.indication
R15
[seqnum not acceptable] /
ER.Sa_DISCONNECT.request
CSL.SAI_DISCONNECT.indication ;
R14
starting a new cycle /
if (ack response timer expired)
SAI.ERROR.report ;
if (FIFO buffer not empty) {
take first message from th buffer;
ER.Sa_DATA.request(
msgtype,userdata,
ackreq,ackresp,
seqnum,ecnum) ;
possibly start ack response timer
}
CONN
Connected
R16
Sa_DATA_indication
(msgtype,userdata,
ackreq,ackresp,seqnum,ecnum) /
R13
SAI_DATA.request(msgtype,userdata) /
[seqnum expected and ecnum valid] /
CSL.SAI_DATA.indication(msgtype,userdata);
[first outgoing msg in this cycle] /
ER.Sa_DATA.request(msgtype,userdata,
ackreq,ackresp,seqnum,ecnum) ;
possibly start ack response timer
[seqnum acceptable and ecnum valid] /
CSL.SAI_DATA.indication(msgtype,userdata);
CSL.SAI_ERROR.report;
[not first outgoing msg in this cycle] /
enqueue in FIFO buffer for later sending
[seqnum old or
(seqnum acceptable and ecnum not valid)] /
CSL.SAI_ERROR.report;