

Additions/modifications to ISO 8802-2 Section 7.9 “Precise description of the Type 2 procedures”

We describe here our additions and modifications to the states, events and actions of ISO Type 2 connection component EFSM. The numberings used here represent the numberings of the ISO 8802 document.

7.9.2 (P.79) Connection service component overview:

The following are a newly added primitive function and a state variable:

- 9) **REJ_TIMER:** This timer is used also when sending SREJ PDU.
- 10) **DL-NL interactions:** Some interactions are changed so that they travel between Connection Component and SAP Component.

7.9.2.1 (P.80) Connection component state descriptions:

The following are newly added states:

- 15) **RSET:** The local connection component has sent an RSET command PDU as the result of detecting an invalid N(R), or the remote station has detected an invalid N(S) in a PDU and has sent an FRMR. Thus the local connection component is in the one-directional sequence number resetting procedure and is waiting for a UA response.
- 16) **RSET_EXPECTED:** The local connection component has detected an invalid N(S) in a PDU and has sent an FRMR. It is expecting an RSET command PDU from the remote station to reset the one-directional sequence number.

7.9.2.2 (P.81) Connection Component event description:

The following events have same names as in ISO, but their semantics are redefined:

- 3) **DATA_REQUEST:** Either (1)as originally defined by ISO or (2)as buffered MULTI-DATA-REQUEST;
Also when this request is used, it is always assumed that SEND_WINDOW_CLOSED=0.
- 30) **RECEIVE_ZZZ_CMD(P=X)_WITH_INVALID_N(R):** ISO description plus SREJ command.
- 31) **RECEIVE_ZZZ_RSP(P=X)_WITH_INVALID_N(R):** ISO description plus SREJ response.

The following are newly added events:

- 48) **RECEIVE_FRMR_RSP(F=X)_REASON=N(S)_ERROR:**
RECEIVE_FRMR_RSP(F=X)_REASON<>N(S)_ERROR: Received an FRMR response, the reason being that N(S) is invalid (=) or of some other reason (<>).
- 49) **RECEIVE_I_XXX:** Received an I command or response PDU.
- 50) **RECEIVE_OTHER_PDU:** Received any other PDU not listed in the transitions for a state.
- 51) **RECEIVE_REJ_XXX:** Received a REJ command or response PDU.
- 52) **RECEIVE_SREJ_XXX:** Received an SREJ command or response PDU.
- 53) **MULTI_DATA_REQUEST:** Upon receiving this high priority request (this has the highest priority of all events), the station component should check immediately whether the PDU could be sent as a multi-destination PDU or not, and respond to the SAP component. If yes, send a MULTI_READY_... (CMD or XXX) to SAP comp. and BUFFER_MULTI_DATA (for retransmission as single-destination PDU); else, send MULTI_NOT_READY to SAP component. And BUFFER_MULTI_DATA (for transmission as single-destination PDU).
- 54) **IS_SEND_WINDOW_OPEN=TRUE:**
IS_SEND_WINDOW_OPEN=FALSE: FALSE if maximum number of outstanding I PDUs reaches K;
TRUE otherwise.
- 55) **IS_SREJ_USED=TRUE:**
IS_SREJ_USED=FALSE: TRUE if Selective Reject scheme is used, which will cause the connection

component to use SREJ PDU instead of REJ PDU. FALSE otherwise.

- 56) **LOCAL_BUSY:** 1 if local busy condition exists; 0 otherwise.

7.9.2.3 (P.83) Connection component action description:

The following actions have same names as in ISO, but their semantics are redefined:

- 41-44) **Start timer actions:** All timers are actually started later by Station Component when PDUs are actually passed to the Physical Layer for transmission. Thus this action here only means that certain information is being sent to the Station Component as to which timer it will later start.

The following are newly added actions:

- 66) **IF_NO_MORE_BUFFERED_DATA_STOP_REJ_TIMER:** If there is no more buffered data for delivery, STOP_REJ_TIMER (REJECT/SREJECT condition is also cleared.)
- 67) **IF_P=1_SEND_R(N)R_RSP(F=1):** If P=1, send an RR response PDU if the station is not busy, or an RNR response PDU if the station is busy.
- 68) **PROCESS_AND_BUFFER_DATA:** When an I PDU with unexpected N(S) is received and IS_SREJ_USED=TRUE, this PDU is buffered for later, orderly delivery; also UPDATE_N(R)_RECEIVED.
- 69) **PROCESS_BUFFERED_DATA:** If the receiving of an I PDU makes any buffered PDU(s) deliverable, process them in the following way:
For each deliverable PDU in sequence do
{ V(R):=V(R)+1;
 DATA_INDICATION;
}
SEND_ACKNOWLEDGE_RSP(F=0); /* Acknowledge the last delivered PDU which implicitly acknowledges all previous ones */
- 70) **RESEND_1_I_CMD:**
RESEND_1_I_RSP:
RESEND_1_I_XXX: These should be used if the remote station uses SREJ (Selective Reject) rather than REJ (Go-Back-N) to resend only a single I PDU requested by the SREJ PDU.
- 71) **RESEND_RSET_CMD:** Resend an RSET command PDU.
- 72) **SEND_R(N)R:** Send an RR command or response PDU if the station is not busy, or an RNR command or response PDU if the station is busy.
- 73) **SEND_SREJ_XXX:** Send one or more SREJ command or response PDU(s) caused by the receiving of an I PDU. These SREJ(s) should inform the sender about all missing PDUs with sequence numbers less than that of the I PDU just received.
- 74) **MULTI_READY_CMD(P=1):**
MULTI_READY_XXX(X=0): See MULTI_DATA_REQUEST for explanation.
- 75) **MULTI_NOT_READY:** See MULTI_DATA_REQUEST for explanation.
- 76) **BUFFER_MULTI_DATA:** See MULTI_DATA_REQUEST for explanation.