5i’

ANNEX D
(to Recommendation Q.921)

State transition table of the point-to-point procedures

of the data link layer

D.1 The state transition table presented in Tables D-1/Q.921 to D-3/Q.921 is based on the eight basic states (see § B.2) recognized in the SDL representation and the related transmitter and receiver conditions.

The state transition table relinquishes to any partitioning of the procedures. It is conceptual and does not prevent a designer from partitioning in his implementation. Moreover, all the processes related to primitive procedures, the management of queues and the exchange of information between adjacent layers are conceptual, not visible from outside of the system and would not impose any constraints on the implementation.

The eight basic states apply to both the transmitter and the receiver within one data link layer entity. However, some of the conditions are confined to the transmitter (e.g. ‘‘peer receiver busy’’), whilst some are confined to the receiver (e.g. ‘‘REJ recovery’’). This implies, if the concept of non-partitioning is adopted, that each transmitter condition has to be combined with each receiver condition resulting in composite states. This state transition table comprises 24 composite states representing the 8 basic states and the related combinations of transmitter and receiver conditions.

Events are defined as follows:

a) primitives;

b) repertoire of frames to be received;

— unnumbered frames (SABME, DISC, UA, DM, UI, FRMR)

— supervisory frames (RR, REJ, RNR)

— information frame (I);

c) internal events (servicing of queues, expiry of timers, receiver busy condition).

The actions to be taken when an event occurs whilst in a specific state comprise:

i) transition to another state

ii) peer-to-peer frame to be transmitted

iii) primitives to be issued

iv) timer actions

v) retry counters

vi) state variables

vii) P/F bit setting

viii) discarding contents of queues.

D.2 *Key to the state transition table*

D.2.1 *Definition of a cell of the state transition table*

Diagram [T17.921], p.

D.2.2 *Key to the contents of a cell*

| Impossible by the definition of the data link layer service.

/ Impossible by the definition of the peer-to-peer data link procedures.

— No action, no state change.

V(S) = V(A) = N(R) Collective term for the two actions V(S) = N(R) and V(A) = N(R).

Timer T200 Start timer T200 if not already running.

TX ACK The acknowledgement of the received I frame may be conveyed by an I frame associated with the information flow in the opposite direction or a supervisory response frame, as appropriate.

‘‘DISCARD’’ Indicates the discarding of the information contained in the information field of the I frame.

(A-O) The codes used in MDL-ERROR-INDICATION signals are defined in Table II-1/Q.921 in Appendix II. When multiple codes are shown, only one applies.

Diagram [T18.921] + Note, p.

blanc

H.T. [1T19.921]

TABLE D-1/Q.921 (1 of 10)
{
State transition table: receiving primitive
}

BASIC STATE	TEI UNASSIGNED	ASSIGN AWAITING TEI	ESTABLISH AWAITING TEI	TEI ASSIGNED
RECEIVER CONDITION				
STATE NUMBER	1	2	3	4
DL-ESTABLISH-REQUEST MDL-ASS-IDN | | | | 3 } RC=0 TX SABME P=1 START T200 5.0 } | | DISC. I QUEUE | | 5.0 }	{ | | | | | | 3 I { | | I I	{ I I		
DL-RELEASE-REQUEST	I	I	I	DL-REL-CONF
DL-DATA-REQUEST	I	I	I	I
I FRAME IN QUEUE V(S)<V(A)+k	I	I	I	I
I FRAME IN QUEUE V(S)=V(A)+k	I	I	I	I
DL-UNIT DATA-REQUEST MDL-ASS-IND UNIT DATA INTO UI QUEUE 2 } | | UNIT DATA INTO UI QUEUE | | }	{ {			
UI FRAME IN QUEUE	I	LEAVE UI FRAME IN QUEUE		TX UI P=0
MDL-ASSIGN-REQUEST STORE TEI VALUE | | | | 4 } STORE TEI VALUE RC=0 TX SABME P=1 START T200 5.0 }	{ | | I	{ | | I	| | I	| | I
MDL-REMOVE-REQUEST | | DISC. UI QUEUE | |	I	I	I	{

| |
1
} {
DL-REL-IND
DISC. UI QUEUE
STOP T200
| |
1
} {
DL-REL-IND
DISC. I and UI
QUEUES
STOP T200
1
} {
DL-REL-CONF
DISC. I and UI
QUEUES
STOP T200
1
} {
DL-REL-CONF
DISC. UI QUEUE
STOP T200
| |
1
}

MDL-ERROR-RESPONSE I DISC. UI QUEUE | | 1 {
DL-REL-IND
DISC. UI QUEUE
1
} I I I I

PERSISTENT DEACTIVATION — {
| |
DISC. UI QUEUE
| |
| |
1
} {
DL-REL-IND
DISC. UI QUEUE
| |
| |
1
} | | DISC. UI QUEUE {
DL-REL-IND
DISC. UI QUEUE
STOP T200
| |
4
} {
DL-REL-IND
DISC. I and UI
QUEUES
STOP T200
4
} {
DL-REL-CONF
DISC. I and UI
QUEUES
STOP T200
4

}	{			
DL-REL-CONF DISC. UI QUEUE STOP T200 | | 4				
}				

Note — The transmitter condition ‘‘pending release’’ may occur only in cases of layer 2 initiated re-establishment.

Table D-1.1/Q.921 (1 of 10) [1T19.921] + Note, p.

TABLE D-1/Q.921 (2 of 10)
{
State transition table: receiving unnumbered frame with correct format
}

BASIC STATE TRANSMITTER CONDITION RECEIVER CONDITION STATE NUMBER	TEI UNASSIGNED 1	ASSIGN AWAITING TEI 2	ESTABLISH AWAITING TEI 3	TEI ASSIGNED 4

{
SABME
P=1
ABLE TO ENTER STATE 7.0
}
DL-EST-IND
V(S,R,A)=0
TX UA F=1
START T203
7.0
	/	/	/	{
				TX DM F=1

{
SABME
P=1
UNABLE TO ENTER STATE 7.0
}
	/	/	/	TX DM F=1

{
SABME
P=0
ABLE TO ENTER STATE 7.0
}
DL-EST-IND
V(S,R,A)=0
TX UA F=0
START T203
7.0
	/	/	/	{
				TX DM F=0

{
SABME
P=0
UNABLE TO ENTER STATE 7.0
}
	/	/	/	TX DM F=0

DISC P=1	/	/	/	TX DM F=1
DISC P=0	/	/	/	TX DM F=0
UA F=1 V(S) = V(A) V(S,R,A)=0 DL-EST-CONF STOP T200 START T203 7.0 } V(S,R,A)=0 STOP T200 START T203 | | 7.0 } DISC I QUEUE RC=0 TX DISC P=1	/	/	/	MDL-ERR-IND(C)

RESTART T200 6 } | | DL-REF-CONF STOP T200 | | 4 }	{			
UA F=1 V(S) ≠ V(A) DISC I QUEUE V(S,R,A)=0 DL-EST-IND STOP T200 START T203 7.0 }	/	/	/	
UA F=0	/	/	/	MDL-ERR-IND(D)
DM F=1 DL-REL-IND STOP T200 | | 4 } DL-REL-IND DISC I QUEUE STOP T200 4 } DL-REL-CONF DISC I QUEUE STOP T200 4 } DL-REL-CONF STOP T200 | | 4 }	/ { { {	/	/	—

{
DM
F=0
ABLE TO ENTER STATE 7.0
}
RC=0
TX SABME P=1
START T200
5.1
}
	/
	—
		/
		—
			/
			—
				{
				—

{
DM
F=0
UNABLE TO ENTER STATE 7.0
}
	/	/	/	—

UI command	/	/	/	DL-UNIT DATA-IND

Table D-1/Q.921 (2 of 10) [2T19.921], p.

H.T. [3T19.921]

TABLE D-1/Q.921 (3 of 10)
{
State transition table: receiving FRMR unnumbered frame with correct format
}

BASIC STATE TRANSMITTER CONDITION RECEIVER CONDITION STATE NUMBER	TEI UNASSIGNED 1	ASSIGN AWAITING TEI 2	ESTABLISH AWAITING TEI 3	TEI ASSIGNED 4	A
FRMR response rejecting SABME	/	/	/	/	
FRMR response rejecting DISC	/	/	/	/	
FRMR response rejecting UA	/	/	/	—	
FRMR response rejecting DM	/	/	/	—	

{
FRMR response
rejecting I command
}
	/	/	/	/	

{
FRMR response
rejecting S frame
}
	/	/	/	/	

FRMR response rejecting FRMR	/	/	/	/	

Table D-1/Q.921 (3 of 10) [3T19.921], p.

H.T. [4T19.921]

TABLE D-1/Q.921 (4 of 10)
{
State transition table: receiving RR supervisory frame with correct format
}

BASIC STATE TRANSMITTER CONDITION RECEIVER CONDITION STATE NUMBER	TEI UNASSIGNED 1	ASSIGN AWAITING TEI 2	ESTABLISH AWAITING TEI 3	TEI ASSIGNED 4	AWA
RR command P=1	/	/	/	—	
RR command P=0	/	/	/	—	
RR response F=0	/	/	/	—	
RR response F=1	/	/	/	—	

Table D-1/Q.921 (4 of 10) [4T19.921], p.

H.T. [5T19.921]

TABLE D-1/Q.921 (5 of 10)
{
State transition table: receiving REJ supervisory frame with correct format
}

BASIC STATE TRANSMITTER CONDITION RECEIVER CONDITION STATE NUMBER	TEI UNASSIGNED 1	ASSIGN AWAITING TEI 2	ESTABLISH AWAITING TEI 3	TEI ASSIGNED 4	AWA
REJ command P=1	/	/	/	—	
REJ command P=0	/	/	/	—	
REJ response F=0	/	/	/	—	
REJ response F=1	/	/	/	—	

Table D-1/Q.921 (5 of 10) [5T19.921], p.

H.T. [6T19.921]

TABLE D-1/Q.921 (6 of 10)
{
State transition table: receiving RNR supervisory frame with correct format
}

BASIC STATE TRANSMITTER CONDITION RECEIVER CONDITION STATE NUMBER	TEI UNASSIGNED 1	ASSIGN AWAITING TEI 2	ESTABLISH AWAITING TEI 3	TEI ASSIGNED 4	AWA
RNR command P=1	/	/	/	—	
RNR command P=0	/	/	/	—	
RNR response F=0	/	/	/	—	
RNR response F=1	/	/	/	—	

Table D-1/Q.921 (6 of 10) [6T19.921], p.

H.T. [7T19.921]

TABLE D-1/Q.921 (7 of 10)
{
State transition table: receiving I command frame with correct format acknowledging all outstanding I frames or containing an N(R) which satisfies V(A) | (= | (R) | (= | (S)
}

BASIC STATE TRANSMITTER CONDITION RECEIVER CONDITION STATE NUMBER	TEI UNASSIGNED 1	ASSIGN AWAITING TEI 2	ESTABLISH AWAITING TEI 3	TEI ASSIGNED 4	AWA

{
I command P=1
N(S) = V(R)
N(R) = V(S)
}
	/	/	/	—	

{
I command P=0
N(S) = V(R)
N(R) = V(S)
}
	/	/	/	—	

{
I command P=1
N(S) ≠ V(R)
N(R) = V(S)
}
	/	/	/	—	

{
I command P=0
N(S) ≠ V(R)
N(R) = V(S)
}
	/	/	/	—	

{
I command P=1
N(S) = V(R)
V(A) | (= | (R) | (= | (S)
}
	/	/	/	—	

{
I command P=0
N(S) = V(R)
V(A) | (= | (R) | (= | (S)
}
	/	/	/	—	

{
I command P=1
N(S) ≠ V(R)
V(A) | (= | (R) | (= | (S)
}
	/	/	/	—	

{
I command P=0
N(S) ≠ V(R)
V(A) | (= | (R) | (= | (S)
}
	/	/	/	—	

Table D-1/Q.921 (7 of 10) [7T19.921], p.

H.T. [8T19.921]

TABLE D-1/Q.921 (8 of 10)
{
State transition table: receiving I command frame with correct format containing an N(R) which satisfies V(A) ≤ N(R) ≤ V(S), or an N(R) error
}

BASIC STATE TRANSMITTER CONDITION RECEIVER CONDITION STATE NUMBER	TEI UNASSIGNED	ASSIGN AWAITING TEI	ESTABLISH AWAITING TEI	TEI ASSIGNED	AWA
	1	2	3	4	

{
I command P=1
N(S) = V(R)
V(A) ≤ N(R) ≤ V(S)
}
	/	/	/	—	

{
I command P=0
N(S) = V(R)
V(A) ≤ N(R) ≤ V(S)
}
	/	/	/	—	

{
I command P=1
N(S) ≠ V(R)
V(A) ≤ N(R) ≤ V(S)
}
	/	/	/	—	

{
I command P=0
N(S) ≠ V(R)
V(A) ≤ N(R) ≤ V(S)
}
	/	/	/	—	

{
I command P=1
N(S) = V(R)
N(R) error
}
	/	/	/	—	

{
I command P=0
N(S) = V(R)
N(R) error
}
	/	/	/	—	

{
I command P=1
N(S) ≠ V(R)
N(R) error
}
	/	/	/	—	

{
I command P=0
N(S) ≠ V(R)
N(R) error
}
	/	/	/	—	

Table D-1/Q.921 (8 of 10) [8T19.921], p.

H.T. [9T19.921]

TABLE D-1/Q.921 (9 of 10)
{
State transition table: Internal events (expiry of timers, receiver busy condition)
}

BASIC STATE TRANSMITTER CONDITION RECEIVER CONDITION STATE NUMBER	TEI UNASSIGNED 1	ASSIGN AWAITING TEI 2	ESTABLISH AWAITING TEI 3	TEI ASSIGNED 4
T200 TIME-OUT RC < | 200 RC=RC+1 TX SABME P=1 START T200 } RC=RC+1 TX DISC P=1 START T200 }	/	/	/ {	/
T200 TIME-OUT RC = N200 | | DL-REL-IND MDL-ERR-IND(G) 4 } DISC I QUEUE DL-REL-IND MDL-ERR-IND(G) 4 } DISC I QUEUE DL-REL-CONF MDL-ERR-IND(G) 4 } | | DL-REL-CONF MDL-ERR-IND(H) 4 }	/ { { {	/	/	/
T203 TIME-OUT	/	/	/	/
SET OWN RECEIVER BUSY (Note)	/	/	/	/

{
CLEAR
OWN RECEIVER BUSY (Note)
}
	/	/	/	/

Note — These signals are generated outside the procedures specified in this state transition table, and may be generated by the connection management entity.

Table D-1/Q.921 (9 of 10) [9T19.921], p.

H.T. [10T19.921]

{
TABLE D-1/Q.921 (10 of 10)
}
{
State transition table: Receiving frame with incorrect format or frame not implemented
}

BASIC STATE TRANSMITTER CONDITION RECEIVER CONDITION STATE NUMBER	TEI UNASSIGNED 1	ASSIGN AWAITING TEI 2	ESTABLISH AWAITING TEI 3	TEI ASSIGNED 4
SABME incorrect length	/	/	/	MDL-ERR-IND(
DISC incorrect length	/	/	/	
UA incorrect length	/	/	/	
DM incorrect length	/	/	/	
FRMR incorrect length	/	/	/	

{
Supervisory frame
RR, REJ, RNR
incorrect length
}
	/	/	/	

N201 error	/	/	/	MDL-ERR-IND(

{
Undefined command and response frames
}
	/	/	/	MDL-ERR-IND(

I field not permitted	/	/	/	MDL-ERR-IND(

Table D-1/Q.921 (10 of 10) [10T19.921], p.

TABLE D-2/Q.921 (1 of 10)

State transition table: receiving primitive

BASIC STATE	MULTIPLE FRAME ESTABLISHED			
TRANSMITTER CONDITION	NORMAL	NORMAL	NORMAL	NORMAL
RECEIVER CONDITION	NORMAL	REJ RECOVERY	OWN REC BUSY	REJ and own REC busy
STATE NUMBER	7.0	7.1	7.2	7.3
DL-ESTABLISH-REQUEST	DISC I QUEUE RC=0 TX SABME P=1 STOP T203 RESTART T200 5.0			
DL-RELEASE-REQUEST	DISC I QUEUE RC=0 TX DISC P=1 STOP T203 RESTART T200 6.0			
DL-DATA-REQUEST	DATA INTO I QUEUE			
I FRAME IN QUEUE V(S)=V(A)+k	TX I P=0 V(S)=V(S)+1 STOP T203 TIMER T200			LEAVE I FRAME IN QUEUE
DL-UNIT DATA-REQUEST	UNIT DATA INTO UI QUEUE			
UI FRAME IN QUEUE	TX UI P=0			
MDL-ASSIGN-REQUEST	I			
MDL-REMOVE-REQUEST	DL-REL-IND DISC I and UI QUEUES STOP T200 STOP T203 1			
MDL-ERROR-RESPONSE	I			
PERSISTENT DEACTIVATION	DL-REL-IND DISC I and UI QUEUES			

STOP T200 STOP T203 4 }					

Table D-2/Q.921 (1 of 10) [1T20.921], p.

TABLE D-2/Q.921 (2 of 10)

{

State transition table: receiving unnumbered frame with correct format

}

BASIC STATE

TRANSMITTER CONDITION	NORMAL	NORMAL	NORMAL	NORMAL	PEER REC BUSY	PEER RE
RECEIVER CONDITION	NORMAL	REJ RECOVERY	OWN REC BUSY	REJ and own REC busy	NORMAL	REJ REC
STATE NUMBER	7.0	7.1	7.2	7.3	7.4	7.
SABME P=1 V(S) = V(A) | |	{					
MDL-ERR-IND(F) V(S,R,A)=0 TX UA F=1 STOP T200 START T203 | |						
}	{					
MDL-ERR-IND(F) V(S,R,A)=0 TX UA F=1 STOP T200 START T203 7.0						
}						
{ SABME P=1 V(S) ≠ V(A) | |						
}	{					
DL-EST-IND MDL-ERR-IND(F) DISC I QUEUE V(S,R,A)=0 TX UA F=1 STOP T200 START T203 | |						
}	{					
DL-EST-IND MDL-ERR-IND(F) DISC I QUEUE V(S,R,A)=0 TX UA F=1 STOP T200 START T203 7.0						
}						
SABME P=0 V(S) = V(A) | |	{					
MDL-ERR-IND(F) V(S,R,A)=0 TX UA F=0 STOP T200 START T203 | |						
}	{					
MDL-ERR-IND(F) V(S,R,A)=0 TX UA F=0 STOP T200 START T203 7.0						
}						
{ SABME P=0 V(S) ≠ V(A) | |						

} DL-EST-IND MDL-ERR-IND(F) DISC I QUEUE V(S,R,A)=0 TX UA F=0 STOP T200 START T203 | | }	{ {					
DL-EST-IND MDL-ERR-IND(F) DISC I QUEUE V(S,R,A)=0 TX UA F=0 STOP T200 START T203 7.0 }						
DISC P=1 | | DL-REL-IND DISC I QUEUE TX UA F=1 STOP T200, T203 4 }	{ 					
DISC P=0 | | DL-REL-IND DISC I QUEUE TX UA F=0 STOP T200, T203 4 }	{ 					

Table D-2/Q.921 (2 of 10) [2T.20.921], p.

{
TABLE D-2/Q.921 (2 of 10 *cont.*
)
}
{
State transition table: receiving unnumbered frame with correct format
}

BASIC STATE	MULTIPLE FRAME ESTABLISHED

TRANSMITTER CONDITION RECEIVER CONDITION STATE NUMBER UA F=1	NORMAL NORMAL 7.0 MDL-ERR-IND(C)	NORMAL REJ RECOVERY 7.1	NORMAL OWN REC BUSY 7.2	NORMAL REJ and own REC busy 7.3	PEER REC BUSY NORMAL 7.4
UA F=0	MDL-ERR-IND(D)				
DM F=1	MDL-ERR-IND(B)				
DM F=0 | | MDL-ERR-IND(E) RC = 0 TX SABME P=1 STOP T203 RESTART T200 5.1 } MDL-ERR-IND(E) RC = 0 TX SABME P=1 RESTART T200 | | 5.1 }	{			{	
UI command	DL-UNIT DATA-IND				

Table D-2/Q.921 (2 of 10 cnt'd) [3T20.921], p.

TABLE D-2/Q.921 (3 of 10)
{
State transition table: receiving FRMR unnumbered frame
with correct format
}

BASIC STATE | MULTIPLE FRAME ESTABLISHED

TRANSMITTER CONDITION RECEIVER CONDITION STATE NUMBER FRMR response rejecting SABME	NORMAL NORMAL 7.0 /	NORMAL REJ RECOVERY 7.1 /	NORMAL OWN REC BUSY 7.2 /	NORMAL REJ and own REC busy 7.3 /	PEER REC BUSY NORMAL 7.4 /	PEER I REJ RI
FRMR response rejecting DISC	/	/	/	/	/	

{
FRMR response
rejecting UA
| |
}
MDL-ERR-IND(K)
RC = 0
TX SABME P=1
STOP T203
RESTART T200
5.1
}
MDL-ERR-IND(K)
RC = 0
TX SABME P=1
RESTART T200
| |
5.1
}
	{			{		

FRMR response rejecting DM	/	/	/	/	/	

{
FRMR response
rejecting I command
| |
}
MDL-ERR-IND(K)
RC = 0
TX SABME P=1
STOP T203
RESTART T200
5.1
}
MDL-ERR-IND(K)
RC = 0
TX SABME P=1
RESTART T200
| |
5.1
}
	{			{		

{
FRMR response
rejecting S frame
}						

FRMR response rejecting FRMR	/	/	/	/	/	

Table D-2/Q.921 (3 of 10) [4T20.921], p.

H.T. [5T20.921]

TABLE D-2/Q.921 (4 of 10)
{
State transition table: receiving RR supervisory frame with correct format
}

BASIC STATE	MULTIPLE FRAME ESTABLISHED

TRANSMITTER CONDITION	NORMAL	NORMAL	NORMAL	NORMAL	PE
RECEIVER CONDITION	NORMAL	REJ RECOVERY	OWN REC BUSY	REJ and own REC busy	
STATE NUMBER	7.0	7.1	7.2	7.3	
{					
RR command					
P=1					
N(R)=V(S)					
| |					
}	{				
TX RR F=1					
STOP T200					
RESTART T203					
V(A)=N(R)					
| |					
}		{			
TX RNR F=1					
STOP T200					
RESTART T203					
V(A)=N(R)					
| |					
}		{			
TX RR F=1					
STOP T200					
START T203					
V(A)=N(R)					
7.0					
}	{				
TX RR F=1					
STOP T200					
START T203					
V(A)=N(R)					
7.1					
}	{				
TX RNR F=1					
STOP T200					
START T203					
V(A)=N(R)					
7.2					
}	{				
TX RNR F=1					
STOP T200					
START T203					
V(A)=N(R)					
7.3					
}					
{					
RR command					
P=0					
N(R)=V(S)					
| |					
}	{				
STOP T200					
RESTART T203					
V(A)=N(R)					
| |					
}				{	
STOP T200					
START T203					
V(A)=N(R)					
7.0					
}	{				
STOP T200					

START T203
V(A)=N(R)
7.1
} {
STOP T200
START T203
V(A)=N(R)
7.2
} {
STOP T200
START T203
V(A)=N(R)
7.3
}

RR response F=0 N(R)=V(S)

{
RR response
F=1
N(R)=V(S)
| |
} {
MDL-ERR-IND(A)
STOP T200
RESTART T203
V(A)=N(R)
| |
} {
MDL-ERR-IND(A)
STOP T200
START T203
V(A)=N(R)
7.0
} {
MDL-ERR-IND(A)
STOP T200
START T203
V(A)=N(R)
7.1
} {
MDL-ERR-IND(A)
STOP T200
START T203
V(A)=N(R)
7.2
} {
MDL-ERR-IND(A)
STOP T200
START T203
V(A)=N(R)
7.3
}

{
RR command
P=1
V(A) N(R) < V(S)
| |
} {
TX RR F=1
RESTART T200
V(A)=N(R)
| |
} {

TX RNR F=1
RESTART T200
V(A)=N(R)
| |
} {
TX RR F=1
RESTART T200
V(A)=N(R)
7.0
} {
TX RR F=1
RESTART T200
V(A)=N(R)
7.1
} {
TX RNR F=1
RESTART T200
V(A)=N(R)
7.2
} {
TX RNR F=1
RESTART T200
V(A)=N(R)
7.3
}

{
RR command
P=0
V(A) N(R) < V(S)
| |
} RESTART T200 V(A)=N(R) | | RESTART

{
RR response
F=0
V(A) N(R) < V(S)
}

{
RR response
F=1
V(A) N(R) < V(S)
| |
} {
MDL-ERR-IND(A)
RESTART T200
V(A)=N(R)
| |
} {
MDL-ERR-IND(A)
RESTART T200
V(A)=N(R)
7.0
} {
MDL-ERR-IND(A)
RESTART T200
V(A)=N(R)
7.1
} {
MDL-ERR-IND(A)
RESTART T200
V(A)=N(R)
7.2
} {

MDL-ERR-IND(A) RESTART T200 V(A)=N(R) 7.3 }					

Table D-2/Q.921 (4 of 10) [5T20.921], p.

H.T. [6T20.921]

{
TABLE D-2/Q.921 (4 of 10 *cont.*
)
}
{
State transition table: receiving RR supervisory frame with correct format
}

BASIC STATE	MULTIPLE FRAME ESTABLISHED

TRANSMITTER CONDITION	NORMAL	NORMAL	NORMAL	NORMAL	PEER REC BUSY
RECEIVER CONDITION	NORMAL	REJ RECOVERY	OWN REC BUSY	REJ and own REC busy	NORMAL
STATE NUMBER	7.0	7.1	7.2	7.3	7.4

{
RR command
P=1
V(A) = N(R) < V(S)
}
	TX RR F=1 | |		TX RNR F=1 | |		TX RR F=1 7.0

{
RR command
P=0
V(A) = N(R) < V(S)
| |
}
	—	—	—	—	

{
RR command
F=0
V(A) = N(R) < V(S)
}
	—	—	—	—	

{
RR response
F=1
V(A) = N(R) < V(S)
}
	MDL-ERR-IND(A) | |				MDL-ERR-IND(A)

{
RR command
P=1
N(R) error
| |
}
	{
	TX RR F=1
	MDL-ERR-IND(J)
	RC =0
	TX SABME P=1
	STOP T203
	RESTART T200
	5.1
	}
		{
		TX RNR F=1
		MDL-ERR-IND(J)
		RC =0
		TX SABME P=1
		STOP T203
		RESTART T200
		5.1
		}			

		{
		TX RR F=1
		MDL-ERR-IND(J)
		RC =0
		TX SABME P=1
		RESTART T200
		| |
		5.1
		}			

		{
		TX RNR F=1
		MDL-ERR-IND(J)
		RC =0
		TX SABME P=1
		RESTART T200
		| |
		5.1
		}			

{
RR command
P=0
N(R) error
| |
}
	{
	MDL-ERR-IND(J)
	RC =0
	TX SABME P=1
	STOP T203
	RESTART T200
	5.1
	}
			{
			MDL-ERR-IND(J)
			RC =0
			TX SABME P=1
			RESTART T200
			| |
			5.1
			}		

RR response F=0 N(R) error					

{
RR response
F=1
N(R) error
| |
}
	{
	MDL-ERR-IND(A)
	MDL-ERR-IND(J)
	RC =0
	TX SABME P=1
	STOP T203
	RESTART T200
	5.1
	}
			{
			MDL-ERR-IND(A)
			MDL-ERR-IND(J)
			RC =0
			TX SABME P=1
			RESTART T200
			| |
			5.1
			}		

Table D-2/Q.921 (4 of 10 cnt'd) [6T20.921], p.

H.T. [7T20.921]

TABLE D-2/Q.921 (5 of 10)
{
State transition table: receiving REJ supervisory frame with correct format
}

BASIC STATE	MULTIPLE FRAME ESTABLISHED

TRANSMITTER CONDITION RECEIVER CONDITION STATE NUMBER	NORMAL NORMAL 7.0	NORMAL REJ RECOVERY 7.1	NORMAL OWN REC BUSY 7.2	NORMAL REJ and own REC busy 7.3	PEER REC BUSY NORMAL 7.4	PEER RE REJ REC 7.
{ REJ command P=1 N(R)=V(S) (Note) | |						
} TX RR F=1 V(A)=N(R) STOP T200 RESTART T203 | |	{					
} TX RNR F=1 V(A)=N(R) STOP T200 RESTART T203 | |		{				
} TX RR F=1 V(A)=N(R) STOP T200 START T203 7.0	{					
} TX RR F=1 V(A)=N(R) STOP T200 START T203 7.1	{					
} TX RNR F=1 V(A)=N(R) STOP T200 START T203 7.2	{					
} TX RNR F=1 V(A)=N(R) STOP T200 START T203 7.3	{					
}						
{ REJ command P=0 N(R)=V(S) (Note) | |						
} V(A)=N(R) STOP T200 RESTART T203 | |	{					
} V(A)=N(R) STOP T200 START T203 7.0				{		
} V(A)=N(R)	{					

STOP T200
START T203
7.1
} {
V(A)=N(R)
STOP T200
START T203
7.2
} {
V(A)=N(R)
STOP T200
START T203
7.3
}

{
REJ response
F=0
N(R)=V(S) (Note)
}

{
REJ response
F=1
N(R)=V(S) (Note)
| |
} {
MDL-ERR-IND(A)
V(A)=N(R)
STOP T200
RESTART T203
| |
} {
MDL-ERR-IND(A)
V(A)=N(R)
STOP T200
START T203
7.0
} {
MDL-ERR-IND(A)
V(A)=N(R)
STOP T200
START T203
7.1
} {
MDL-ERR-IND(A)
V(A)=N(R)
STOP T200
START T203
7.2
} {
MDL-ERR-IND(A)
V(A)=N(R)
STOP T200
START T203
7.3
}

{
REJ command
P=1
V(A) N(R) < V(S)
| |
} {
TX RR F=1

V(S)=V(A)=N(R)
STOP T200
START T203
| |
} {
TX RNR F=1
V(S)=V(A)=N(R)
STOP T200
START T203
| |
} {
TX RR F=1
V(S)=V(A)=N(R)
STOP T200
START T203
7.0
} {
TX RR F=1
V(S)=V(A)=N(R)
STOP T200
START T203
7.1
} {
TX RNR F=1
V(S)=V(A)=N(R)
STOP T200
START T203
7.2
} {
TX RNR F=1
V(S)=V(A)=N(R)
STOP T200
START T203
7.3
}

{
REJ command
P=0
V(A) N(R) < V(S)
| |
} {
V(S)=V(A)=N(R)
STOP T200
START T203
| |
} {
V(S)=V(A)=N(R)
STOP T200
START T203
7.0
} {
V(S)=V(A)=N(R)
STOP T200
START T203
7.1
} {
V(S)=V(A)=N(R)
STOP T200
START T203
7.2
} {
V(S)=V(A)=N(R)

STOP T200
START T203
7.3
}

---

lw(54p) | lw(36p) | cw(36p) | lw(36p) | cw(36p) | lw(36p) | lw(36p) | lw(36p) | lw(36p) .
{
REJ response
F=0
V(A) N(R) < V(S)
}

---

lw(54p) | lw(36p) | cw(36p) | lw(36p) | cw(36p) | lw(36p) | lw(36p) | lw(36p) | lw(36p) .
{
REJ response
F=1
V(A) N(R) < V(S)
| |
} {
MDL-ERR-IND(A)
V(S)=V(A)=N(R)
STOP T200
START T203
| |
} {
MDL-ERR-IND(A)
V(S)=V(A)=N(R)
STOP T200
START T203
7.0
} {
MDL-ERR-IND(A)
V(S)=V(A)=N(R)
STOP T200
START T203
7.1
} {
MDL-ERR-IND(A)
V(S)=V(A)=N(R)
STOP T200
START T203
7.2
} {
MDL-ERR-IND(A)
V(S)=V(A)=N(R)
STOP T200
START T203
7.3
}

Note —
This event is impossible by the definition of the peer-to-peer data link procedures. However, it would not harm the information transfer, if actions according to this table are taken.

Table D-2/Q.921 (5 of 10) [7T20.921], p.

H.T. [8T20.921]

center
box;
cw(342p)
.

{
TABLE D-2/Q.921 (5 of 10
cont.
)

}
cw(342p)
.

{
State transition table: receiving REJ supervisory frame with correct for-mat

}

BASIC STATE	MULTIPLE FRAME ESTABLISHED					

TRANSMITTER CONDITION	NORMAL	NORMAL	NORMAL	NORMAL	PEER REC BUSY	PEER RE
RECEIVER CONDITION	NORMAL	REJ RECOVERY	OWN REC. BUSY	REJ and own REC. busy	NORMAL	REJ REC
STATE NUMBER	7.0	7.1	7.2	7.3	7.4	7
{ REJ command P=1 N(R) error | |						
} TX RR F=1 MDL-ERR-IND(J) RC =0 TX SABME P=1 STOP T203 RESTART T200 5.1	{ 					
} TX RNR F=1 MDL-ERR-IND(J) RC =0 TX SABME P=1 STOP T203 RESTART T200 5.1		{ 				
} TX RR F=1 MDL-ERR-IND(J) RC =0 TX SABME P=1 RESTART T200 | | 5.1		{ 				
} TX RNR F=1 MDL-ERR-IND(J) RC =0 TX SABME P=1 RESTART T200 | | 5.1		{ 				
}						
{ REJ command P=0 N(R) error | |						
} MDL-ERR-IND(J) RC =0 TX SABME P=1 STOP T203 RESTART T200 5.1	{ 					
} MDL-ERR-IND(J) RC =0 TX SABME P=1 RESTART T200 | | 5.1				{ 		
}						
REJ response F=0 N(R) error						

{
REJ response
F=1
N(R) error
| |
}
	{
	MDL-ERR-IND(A)
	MDL-ERR-IND(J)
	RC =0
	TX SABME P=1
	STOP T203
	RESTART T200
	5.1
	}
				{
				MDL-ERR-IND(A)
				MDL-ERR-IND(J)
				RC =0
				TX SABME P=1
				RESTART T200
				| |
				5.1
				}		

Table D-2/Q.921 (5 of 10 cnt'd) [8T20.921], p.

H.T. [9T20.921]

TABLE D-2/Q.921 (6 of 10)
{
State transition table: receiving RNR supervisory frame with correct format
}

BASIC STATE	MULTIPLE FRAME ESTABLISHED

TRANSMITTER CONDITION	NORMAL	NORMAL	NORMAL	
RECEIVER CONDITION	NORMAL	REJ RECOVERY	OWN REC BUSY	REJ
STATE NUMBER	7.0	7.1	7.2	
{				
RNR command P=1 N(R)=V(S) | |				
}	{			
TX RR F=1 STOP T203 RESTART T200 V(A)=N(R) 7.4				
}	{			
TX RR F=1 STOP T203 RESTART T200 V(A)=N(R) 7.5				
}	{			
TX RNR F=1 STOP T203 RESTART T200 V(A)=N(R) 7.6				
}	{			
TX RNR F=1 STOP T203 RESTART T200 V(A)=N(R) 7.7				
}	{			
TX RR F=1 RESTART T200 V(A)=N(R) | | | |				
}		{		
TX RNR F=1 RESTART T200 V(A)=N(R) | | | |				
}				
{				
RNR command P=0 N(R)=V(S) | |				
}	{			
STOP T203 RESTART T200 V(A)=N(R) 7.4				
}	{			
STOP T203 RESTART T200 V(A)=N(R) 7.5				
}	{			
STOP T203				

RESTART T200
V(A)=N(R)
7.6
}
{
STOP T203
RESTART T200
V(A)=N(R)
7.7
}
{
RESTART T200
V(A)=N(R)
| |
| |
}

RNR response F=0 N(R)=V(S)

{
RNR response
F=1
N(R)=V(S)
| |
}
{
MDL-ERR-IND(A)
STOP T203
RESTART T200
V(A)=N(R)
7.4
}
{
MDL-ERR-IND(A)
STOP T203
RESTART T200
V(A)=N(R)
7.5
}
{
MDL-ERR-IND(A)
STOP T203
RESTART T200
V(A)=N(R)
7.6
}
{
MDL-ERR-IND(A)
STOP T203
RESTART T200
V(A)=N(R)
7.7
}
{
MDL-ERR-IND(A)
RESTART T200
V(A)=N(R)
| |
7.6
}

{
RNR command
P=1
V(A) N(R) < V(S)
| |
}
{
TX RR F=1
RESTART T200
V(A)=N(R)
7.4
}
{

TX RR F=1
RESTART T200
V(A)=N(R)
7.5
}
{
TX RNR F=1
RESTART T200
V(A)=N(R)
7.6
}
{
TX RNR F=1
RESTART T200
V(A)=N(R)
7.7
}
{
TX RR F=1
RESTART T200
V(A)=N(R)
| |
}
{
TX RNR F=1
RESTART T200
V(A)=N(R)
| |
}

{
RNR command
P=0
V(A) N(R) < V(S)
| |
}
RESTART T200 V(A)=N(R) 7.4
RESTART T200 V(A)=N(R) 7.5
RESTART T200 V(A)=N(R) 7.6
RESTAR

{
RNR response
F=0
V(A) N(R) < V(S)
}

{
RNR response
F=1
V(A) N(R) < V(S)
| |
}
{
MDL-ERR-IND(A)
RESTART T200
V(A)=N(R)
7.4
}
{
MDL-ERR-IND(A)
RESTART T200
V(A)=N(R)
7.5
}
{
MDL-ERR-IND(A)
RESTART T200
V(A)=N(R)
7.6
}
{
MDL-ERR-IND(A)
RESTART T200
V(A)=N(R)
7.7
}
{

MDL-ERR-IND(A) RESTART T200 V(A)=N(R) | | }				

Table D-2/Q.921 (6 of 10) [9T20.921], p.

H.T. [10T20.921]

{
TABLE D-2/Q.921 (6 of 10 *cont.*
)
}
{
State transition table: receiving RNR supervisory frame with correct format
}

BASIC STATE	MULTIPLE FRAME ESTABLISHED

TRANSMITTER CONDITION	NORMAL	NORMAL	NORMAL	NORMAL	PEER REC BUSY	PEER RE
RECEIVER CONDITION	NORMAL	REJ RECOVERY	OWN REC BUSY	REJ and own REC busy	NORMAL	REJ REC
STATE NUMBER	7.0	7.1	7.2	7.3	7.4	7.
{ RNR command P=1 N(R) error | | }	{ TX RR F=1 MDL-ERR-IND(J) RC =0 TX SABME P=1 STOP T203 RESTART T200 5.1 }	{ TX RNR F=1 MDL-ERR-IND(J) RC =0 TX SABME P=1 STOP T203 RESTART T200 5.1 }	{ TX RR F=1 MDL-ERR-IND(J) RC =0 TX SABME P=1 RESTART T200 | | 5.1 }	{ TX RNR F=1 MDL-ERR-IND(J) RC =0 TX SABME P=1 RESTART T200 | | 5.1 }		
{ RNR command P=0 N(R) error | | }	{ MDL-ERR-IND(J) RC =0 TX SABME P=1 STOP T203 RESTART T200 5.1 }			{ MDL-ERR-IND(J) RC =0 TX SABME P=1 RESTART T200 | | 5.1 }		
RNR response F=0 N(R) error						

{
RNR response
F=1
N(R) error
| |
}
	{
	MDL-ERR-IND(A)
	MDL-ERR-IND(J)
	RC =0
	TX SABME P=1
	STOP T203
	RESTART T200
	5.1
	}
				{
				MDL-ERR-IND(A)
				MDL-ERR-IND(J)
				RC =0
				TX SABME P=1
				RESTART T200
				| |
				5.1
				}		

Table D-2/Q.921 (6 of 10 cnt'd) [10T20.921], p.

TABLE D-2/Q.921 (7 of 10)
{
State transition table: receiving I command frame with correct format acknowledging all outstanding I frames or containing an N(R) which satisfies V(A) | | (R) | | (S)
}

BASIC STATE	MULTIPLE FRAME ESTABLISHED					
TRANSMITTER CONDITION	NORMAL	NORMAL	NORMAL	NORMAL	PEER REC BUSY	PEER REC
RECEIVER CONDITION	NORMAL	REJ RECOVERY	OWN REC BUSY	REJ and own REC busy	NORMAL	REJ RECC
STATE NUMBER	7.0	7.1	7.2	7.3	7.4	7.5
{ I command P=1 N(S) = V(R) N(R) = V(S)						
| | }	{					
V(R)=V(R)+1 DL=DATA=IND TX RR F=1 STOP T200 RESTART T203 V(A)=N(R)						
| | }	{					
V(R)=V(R)+1 DL=DATA=IND TX RR F=1 STOP T200 RESTART T203 V(A)=N(R) 7.0						
} ‘‘DISCARD’’ TX RNR F=1 STOP T200 RESTART T203 V(A)=N(R)	{					
| | | | }		{				
V(R)=V(R)+1 DL-DATA-IND TX RR F=1 V(A)=N(R)						
| | | | | | }	{					
V(R)=V(R)+1 DL-DATA-IND TX RR F=1 V(A)=N(R)						
| | | | 7.4						
} ‘‘DISCARD’’ TX RNR F=1	{					

V(A)=N(R)
| |
| |
| |
| |
}

{
I command P=0
N(S) = V(R)
N(R) = V(S)
| |
} {
V(R)=V(R)+1
DL=DATA=IND
TX ACK
STOP T200
RESTART T203
V(A)=N(R)
| |
} {
V(R)=V(R)+1
DL=DATA=IND
TX ACK
STOP T200
RESTART T203
V(A)=N(R)
7.0
} {
‘‘DISCARD’’
STOP T200
RESTART T203
V(A)=N(R)
| |
| |
| |
} {
V(R)=V(R)+1
DL-DATA-IND
TX RR F=0
V(A)=N(R)
| |
| |
| |
} {
V(R)=V(R)+1
DL-DATA-IND
TX RR F=0
V(A)=N(R)
| |
| |
7.4
} {
‘‘DISCARD’’
V(A)=N(R)
| |
| |
| |
| |
| |
}

{
I command P=1

N(S) ≠ V(R)
N(R) = V(S)
| |
} {
‘‘DISCARD’’
TX REJ F=1
STOP T200
RESTART T203
V(A)=N(R)
7.1
} {
‘‘DISCARD’’
TX RR F=1
STOP T200
RESTART T203
V(A)=N(R)
| |
} {
‘‘DISCARD’’
TX RNR F=1
STOP T200
RESTART T203
V(A)=N(R)
| |
} {
‘‘DISCARD’’
TX REJ F=1
V(A)=N(R)
| |
| |
7.5
} {
‘‘DISCARD’’
TX RR F=1
V(A)=N(R)
| |
| |
| |
} {
‘‘DISCARD’’
TX RNR F=1
V(A)=N(R)
| |
| |
| |
}

{
I command P=0
N(S) ≠ V(R)
N(R) = V(S)
| |
} {
‘‘DISCARD’’
TX REJ F=1
STOP T200
RESTART T203
V(A)=N(R)
7.1
} {
‘‘DISCARD’’
STOP T200
RESTART T203

V(A)=N(R)
| |
| |
} {
‘‘DISCARD’’
TX REJ F=0
V(A)=N(R)
| |
| |
7.5
} {
‘‘DISCARD’’
V(A)=N(R)
| |
| |
| |
| |
}

lw(54p) | lw(36p) | lw(36p) | lw(36p) | cw(36p) | lw(36p) | lw(36p) | lw(36p) | cw(36p) .
{
I command P=1
N(S) = V(R)
V(A) | | (R) | | (S)
| |
} {
V(R)=V(R)+1
DL=DATA=IND
TX RR F=1
RESTART T200
V(A)=N(R)
| |
} {
V(R)=V(R)+1
DL=DATA=IND
TX RR F=1
RESTART T200
V(A)=N(R)
7.0
} {
‘‘DISCARD’’
TX RNR F=1
RESTART T200
V(A)=N(R)
| |
| |
} {
V(R)=V(R)+1
DL-DATA-IND
TX RR F=1
V(A)=N(R)
| |
| |
} {
V(R)=V(R)+1
DL-DATA-IND
TX RR F=1
V(A)=N(R)
| |
7.4
} {
‘‘DISCARD’’
TX RNR F=1

V(A)=N(R)
| |
| |
| |
}

---

lw(54p) | lw(36p) | lw(36p) | lw(36p) | cw(36p) | lw(36p) | lw(36p) | lw(36p) | cw(36p) .
{
I command P=0
N(S) = V(R)
V(A) | | (R) | | (S)
| |
} {
V(R)=V(R)+1
DL=DATA=IND
TX ACK
RESTART T200
V(A)=N(R)
| |
} {
V(R)=V(R)+1
DL=DATA=IND
TX ACK
RESTART T200
V(A)=N(R)
7.0
} {
‘‘DISCARD’’
RESTART T200
V(A)=N(R)
| |
| |
| |
} {
V(R)=V(R)+1
DL-DATA-IND
TX RR F=0
V(A)=N(R)
| |
| |
} {
V(R)=V(R)+1
DL-DATA-IND
TX RR F=0
V(A)=N(R)
| |
7.4
} {
‘‘DISCARD’’
V(A)=N(R)
| |
| |
| |
| |
}

---

lw(54p) | lw(36p) | lw(36p) | lw(36p) | cw(36p) | lw(36p) | lw(36p) | lw(36p) | cw(36p) .
{
I command P=1
N(S) ≠ V(R)
V(A) | | (R) | | (S)
| |
} {
‘‘DISCARD’’

TX REJ F=1
RESTART T200
V(A)=N(R)
7.1
} {
‘‘DISCARD’’
TX RR F=1
RESTART T200
V(A)=N(R)
| |
} {
‘‘DISCARD’’
TX RNR F=1
RESTART T200
V(A)=N(R)
| |
} {
‘‘DISCARD’’
TX REJ F=1
V(A)=N(R)
| |
7.5
} {
‘‘DISCARD’’
TX RR F=1
V(A)=N(R)
| |
| |
} {
‘‘DISCARD’’
TX RNR F=1
V(A)=N(R)
| |
| |
}

---

lw(54p) | lw(36p) | lw(36p) | lw(36p) | cw(36p) | lw(36p) | lw(36p) | lw(36p) | cw(36p) .
{
I command P=0
N(S) ≠ V(R)
V(A) | | (R) | | (S)
| |
} {
‘‘DISCARD’’
TX REJ F=0
RESTART T200
V(A)=N(R)
7.1
} {
‘‘DISCARD’’
RESTART T200
V(A)=N(R)
| |
| |
} {
‘‘DISCARD’’
TX REJ F=0
V(A)=N(R)
| |
7.5
} {
‘‘DISCARD’’
V(A)=N(R)

}						

Table D-2/Q.921 (7 of 10) [11T20.921], p.

H.T. [12T20.921]

TABLE D-2/Q.921 (8 of 10)
{
State transition table: receiving I command frame with correct format containing an N(R) which satisfies V(A) ≤ N(R) ≤ V(S), or an N(R) error

}

BASIC STATE	MULTIPLE FRAME ESTABLISHED

TRANSMITTER CONDITION	NORMAL	NORMAL	NORMAL	NORMA
RECEIVER CONDITION	NORMAL	REJ RECOVERY	OWN REC BUSY	REJ and own R
STATE NUMBER	7.0	7.1	7.2	7.3
{				
I command P=1				
N(S) = V(R)				
V(A)=N(R) | | (S)				
}	{			
V(R)=V(R)+1				
DL=DATA=IND				
TX RR F=1				
| |				
}	{			
V(R)=V(R)+1				
DL=DATA=IND				
TX RR F=1				
7.0				
}	{			
‘‘DISCARD’’				
TX RNR F=1				
| |				
| |				
}		{		
V(R)=V(R)+1				
DL-DATA-IND				
TX RR F=1				
| |				
}	{			
V(R)=V(R)+1				
DL-DATA-IND				
TX RR F=1				
7.4				
}	{			
‘‘DISCARD’’				
TX RNR F=1				
| |				
| |				
}				
{				
I command P=0				
N(S) = V(R)				
V(A)=N(R) | | (S)				
}	{			
V(R)=V(R)+1				
DL=DATA=IND				
TX ACK				
| |				
}	{			
V(R)=V(R)+1				
DL=DATA=IND				
TX ACK				
7.0				
}	{			
‘‘DISCARD’’				
| |				
| |				
| |				
}		{		
V(R)=V(R)+1				
DL-DATA-IND				
TX RR F=0				
| |				

}	{		
V(R)=V(R)+1 DL-DATA-IND TX RR F=0 7.4			
}	{		
‘‘DISCARD’’ | | | | | |			
}			
{			
I command P=1 N(S) ≠ V(R) V(A)=N(R) | | (S)			
}	‘‘DISCARD’’ TX REJ F=1 7.1	‘‘DISCARD’’ TX RR F=1 | |	{
‘‘DISCARD’’ TX RNR F=1 | |			
}		‘‘DISCARD’’ TX REJ F=1 7.5	‘‘DISCARD’’ TX RR F=1 | | {
‘‘DISCARD’’ TX RNR F=1 | |			
}			
{			
I command P=0 N(S) ≠ V(R) V(A)=N(R) | | (S)			
}	‘‘DISCARD’’ TX REJ F=0 7.1	‘‘DISCARD’’ | | | |	
{			
I command P=1 N(S) = V(R) N(R) error | |			
}	{		
V(R)=V(R)+1 DL=DATA=IND TX RR F=1 MDL-ERR-INJ(J) RC = 0 TX SABME P=1 STOP T203 RESTART T200 5.1			
}		{	
‘‘DISCARD’’ TX RNR F=1 MDL-ERR-INJ(J) RC = 0 TX SABME P=1 STOP T203 RESTART T200 | |			
5.1			
}		{	
V(R)=V(R)+1 DL=DATA=IND TX RR F=1 MDL-ERR-INJ(J) RC = 0 TX SABME P=1 RESTART T200			

| |
5.1
} {
‘‘DISCARD’’
TX RNR F=1
MDL-ERR-INJ(J)
RC = 0
TX SABME P=1
RESTART T200
| |
| |
5.1
}

{
I command P=0
N(S) = V(R)
N(R) error
| |
} {
V(R)=V(R)+1
DL=DATA=IND
MDL-ERR-INJ(J)
RC = 0
TX SABME P=1
STOP T203
RESTART T200
5.1
} {
‘‘DISCARD’’
MDL-ERR-INJ(J)
RC = 0
TX SABME P=1
STOP T203
RESTART T200
| |
5.1
} {
V(R)=V(R)+1
DL=DATA=IND
MDL-ERR-INJ(J)
RC = 0
TX SABME P=1
RESTART T200
| |
5.1
} {
‘‘DISCARD’’
MDL-ERR-INJ(J)
RC = 0
TX SABME P=1
RESTART T200
| |
| |
5.1
}

{
I command P=1
N(S) ≠ V(R)
N(R) error
| |
} {
‘‘DISCARD’’

TX REJ F=1
MDL-ERR-IND(J)
RC = 0
TX SABME P=1
STOP T203
RESTART T200
5.1
} {
‘‘DISCARD’’
TX RR F=1
MDL-ERR-IND(J)
RC = 0
TX SABME P=1
STOP T203
RESTART T200
5.1
} {
‘‘DISCARD’’
TX RNR F=1
MDL-ERR-IND(J)
RC = 0
TX SABME P=1
STOP T203
RESTART T200
5.1
} {
‘‘DISCARD’’
TX REJ F=1
MDL-ERR-IND(J)
RC = 0
TX SABME P=1
RESTART T200
| |
5.1
} {
‘‘DISCARD’’
TX RR F=1
MDL-ERR-IND(J)
RC = 0
TX SABME P=1
RESTART T200
| |
5.1
} {
‘‘DISCARD’’
TX RNR F=1
MDL-ERR-IND(J)
RC = 0
TX SABME P=1
RESTART T200
| |
5.1
}

---

lw(54p) | lw(36p) | lw(36p) | lw(36p) | cw(36p) | lw(36p) | lw(36p) | lw(36p) | cw(36p) .
{
I command P=0
N(S) ≠ V(R)
N(R) error
| |
} {
‘‘DISCARD’’
TX REJ F=0

MDL-ERR-IND(J)
RC = 0
TX SABME P=1
STOP T203
RESTART T200
5.1
}
	{
	‘‘DISCARD’’
	MDL-ERR-IND(J)
	RC = 0
	TX SABME P=1
	STOP T203
	RESTART T200
	| |
	5.1
	}
			{
			‘‘DISCARD’’
			TX REJ F=0
			MDL-ERR-IND(J)
			RC = 0
			TX SABME P=1
			RESTART T200
			| |
			5.1
			}
				{
				‘‘DISCARD’’
				MDL-ERR-IND(J)
				RC = 0
				TX SABME P=1
				RESTART T200
				| |
				| |
				5.1
				}

Table D-2/Q.921 (8 of 10) [12T20.921], p.

H.T. [13T20.921]

TABLE D-2/Q.921 (9 of 10)
{
State transition table: internal events (expiry of timers, receiver busy condition)

}

BASIC STATE	MULTIPLE FRAME ESTABLISHED			
TRANSMITTER CONDITION	NORMAL	NORMAL	NORMAL	NOR
RECEIVER CONDITION	NORMAL	REJ RECOVERY	OWN REC BUSY	REJ and ov
STATE NUMBER	7.0	7.1	7.2	7

{
T200 TIME-OUT
RC < | 200
| |
}
	{			

RC=0 either V(S)=V(S)—1 TX I P=1 V(S)=V(S)+1 or TX RR P=1 then RC=RC+1 START T200 8.0				
}	{			
RC=0 either V(S)=V(S)—1 TX I P=1 V(S)=V(S)+1 or TX RR P=1 then RC=RC+1 START T200 8.1				
}	{			
RC=0 either V(S)=V(S)—1 TX I P=1 V(S)=V(S)+1 or TX RNR P=1 then RC=RC+1 START T200 8.2				
}	{			
RC=0 either V(S)=V(S)—1 TX I P=1 V(S)=V(S)+1 or TX RNR P=1 then				

RC=RC+1 START T200 8.3 } RC=0 TX RR P=1 RC=RC+1 START T200 8.4 } RC=0 TX RR P=1 RC=RC+1 START T200 8.5 } RC=0 TX RNR P=1 RC=RC+1 START T200 8.6 } RC=0 TX RNR P=1 RC=RC+1 START T200 8.7 }	{ { { {			

{
T200 TIME-OUT
RC = N200
| |
}
	/	/	/	

T203 TIME-OUT | | RC=0 TX RNR P=1 START T200 8.2 } RC=0 TX RNR P=1 START T200 8.3 }	RC=0 TX RR P=1 START T200 8.0 { /	RC=0 TX RR P=1 START T200 8.1 /	{ /	/
SET OWN RECEIVER BUSY (Note)	TX RNR F=0 7.2	TX RNR F=0 7.3	—	

{
CLEAR
OWN RECEIVER BUSY (Note)
}
TX RR F=0
7.5
}
	—	—	TX RR F=0 7.0	TX RR F=

Note — These signals are generated outside the procedures specified in this state transition table, and may be generated by the connection management entity.

Table D-2/Q.921 (9 of 10) [13T20.921], p.

H.T. [14T20.921]

{
TABLE D-2/Q.921 (10 of 10)
}
{
State transition table: receiving frame with incorrect format or frame not implemented
}

BASIC STATE |	MULTIPLE FRAME ESTABLISHED					
TRANSMITTER CONDITION RECEIVER CONDITION STATE NUMBER	NORMAL NORMAL 7.0	NORMAL REJ RECOVERY 7.1	NORMAL OWN REC BUSY 7.2	NORMAL REJ and own REC busy 7.3	PEER REC BUSY NORMAL 7.4	P F
SABME incorrect length | | MDL-ERR-IND(N) RC = 0 TX SABME P=1 STOP T203 RESTART T200 5.1	{					
DISC incorrect length						
UA incorrect length						
DM incorrect length						
FRMR incorrect length						

{
Supervisory frame
RR, REJ, RNR
incorrect length
}						

N201 error | | MDL-ERR-IND(0) RC = 0 TX SABME P=1 STOP T203 RESTART T200 5.1	{					

{
Undefined command and response frames
| |
}
	{					

MDL-ERR-IND(L) RC = 0 TX SABME P=1 STOP T203 RESTART T200 5.1						
I field not permitted | | MDL-ERR-IND(M) RC = 0 TX SABME P=1 STOP T203 RESTART T200 5.1	{					

Table D-2/Q.921 (10 of 10) [14T20.921], p.

TABLE D-3/Q.921 (1 of 10)
{
State transition table: receiving primitive
}

BASIC STATE	TIMER RECOVERY

TRANSMITTER CONDITION RECEIVER CONDITION STATE NUMBER	NORMAL NORMAL 8.0	NORMAL REJ RECOVERY 8.1	NORMAL OWN REC BUSY 8.2	NORMAL REJ and own REC busy 8.3	PEER N

{
DL-ESTABLISH-REQUEST
| |
}
DISC. I QUEUE
RC=0
TX SABME P=1
RESTART T200
5.0
}
	{				

{
DL-RELEASE-REQUEST
| |
}
DISC. I QUEUE
RC=0
TX DISC P=1
RESTART T200
6
}
	{				

DL-DATA-REQUEST	DATA INTO I QUEUE				

{
I FRAME IN QUEUE
V(S) | | (A)+k
}
	LEAVE I FRAME IN QUEUE				

I FRAME IN QUEUE V(S)=V(A)+k					
DL-UNIT DATA-REQUEST	UNIT DATA INTO UI QUEUE				
UI FRAME IN QUEUE	TX UI P=0				
MDL-ASSIGN-REQUEST	I				

{
MDL-REMOVE-REQUEST
| |
}
DL-REL-IND
DISC. I and UI QUEUES
STOP T200
1
}
	{				

MDL-ERROR-RESPONSE	I				
PERSISTENT DEACTIVATION | | DL-REL-IND DISC. I and UI QUEUES STOP T200 4 }	{				

Table D-3/Q.921 (1 of 10) [1T21.921], p.

TABLE D-3/Q.921 (2 of 10)
{
State transition table: receiving unnumbered frame with correct format
}

BASIC STATE	TIMER RECOVERY

TRANSMITTER CONDITION	NORMAL	NORMAL	NORMAL	NORMAL	PEER REC BUSY	PEER RE
RECEIVER CONDITION	NORMAL	REJ RECOVERY	OWN REC BUSY	REJ and own REC busy	NORMAL	REJ REC
STATE NUMBER	8.0	8.1	8.2	8.3	8.4	8.
SABME P=1 V(S) = V(A) | |	{					
MDL-ERR-IND(F) V(S,R,A)=0 TX UA F=1 STOP T200 START T203 7.0 }						

{
SABME
P=1
V(S) ≠ V(A)
| |
}
	{					

DL-EST-IND MDL-ERR-IND(F) DISC. I QUEUE V(S,R,A)=0 TX UA F=1 STOP T200 START T203 7.0 }						
SABME P=0 V(S) = V(A) | |	{					
MDL-ERR-IND(F) V(S,R,A)=0 TX UA F=0 STOP T200 START T203 7.0 }						

{
SABME
P=0
V(S) ≠ V(A)
| |
}
	{					

DL-EST-IND MDL-ERR-IND(F) DISC. I QUEUE V(S,R,A)=0 TX UA F=0 STOP T200 START T203 7.0 }						
DISC P=1 | |	{					
DL-REL-IND DISC. I QUEUE TX UA F=1 STOP T200 4 }						
DISC P=0 | |	{					
DL-REL-IND DISC. I QUEUE TX UA F=0 STOP T200 4						

}						

Table D-3/Q.921 (2 of 10) [2T21.921], p.

H.T. [3T21.921]

{
TABLE D-3/Q.921 (2 of 10 *cont.*
)
}
{
State transition table: receiving unnumbered frame with correct format
}

BASIC STATE	TIMER RECOVERY

TRANSMITTER CONDITION RECEIVER CONDITION STATE NUMBER UA F=1	NORMAL NORMAL 8.0 MDL-ERR-IND(C)	NORMAL REJ RECOVERY 8.1	NORMAL OWN REC BUSY 8.2	NORMAL REJ and own REC busy 8.3	PEER REC BUSY NORMAL 8.4
UA F=0	MDL-ERR-IND(D)				
DM F=1 | | MDL-ERR-IND(B) RC = 0 TX SABME P=1 RESTART T200 5.1 }	{				
DM F=0 | | MDL-ERR-IND(E) RC = 0 TX SABME P=1 RESTART T200 5.1 }	{				
UI command	DL-UNIT DATA-IND				

Table D-3/Q.921 (2 of 10 cnt'd) [3T21.921], p.

H.T. [4T21.921]

TABLE D-3/Q.921 (3 of 10)
{
State transition table: receiving FRMR unnumbered frame with correct format
}

BASIC STATE	TIMER RECOVERY

TRANSMITTER CONDITION RECEIVER CONDITION STATE NUMBER FRMR response rejecting SABME	NORMAL NORMAL 8.0 /	NORMAL REJ RECOVERY 8.1 /	NORMAL OWN REC BUSY 8.2 /	NORMAL REJ and own REC busy 8.3 /	PEER REC BUSY NORMAL 8.4 /	PEER REJ R
FRMR response rejecting DISC	/	/	/	/	/	
FRMR response rejecting UA	/	/	/	/	/	
FRMR response rejecting DM	/	/	/	/	/	

{
FRMR response
rejecting I command
| |
}
MDL-ERR-IND(K)
RC = 0
TX SABME P=1
RESTART T200
5.1
}
	{					

{
FRMR response
rejecting S frame
}						

FRMR response rejecting FRMR	/	/	/	/	/	

Table D-3/Q.921 (3 of 10) [4T21.921], p.

H.T. [5T21.921]

TABLE D-3/Q.921 (4 of 10)
{
State transition table: receiving RR supervisory frame with correct format, clearance of timer recovery if there is F = 1 only
}

BASIC STATE	TIMER RECOVERY

TRANSMITTER CONDITION RECEIVER CONDITION STATE NUMBER	NORMAL NORMAL 8.0	NORMAL REJ RECOVERY 8.1	NORMAL OWN REC BUSY 8.2	NORMAL REJ and own REC busy 8.3

{
RR command
P=1
V(A) N(R) V(S)
| |
}
	TX RR F=1 V(A)=N(R) | |		TX RNR F=1 V(A)=N(R) | |	TX

{
RR command
P=0
V(A) N(R) V(S)
}
	V(A)=N(R) | |			V(A

{
RR response
F=0
V(A) N(R) V(S)
}				

{
RR response
F=1
V(A) N(R) V(S)
| |
}
	{
	V(S)=N(R)
	STOP T200
	START T203
	V(A)=N(R)
	7.0
	}
		{
		V(S)=N(R)
		STOP T200
		START T203
		V(A)=N(R)
		7.1
		}
			{
			V(S)=N(R)
			STOP T200
			START T203
			V(A)=N(R)
			7.2
			}
				{
				V(S)=N(R)
				STOP T200
				START T203
				V(A)=N(R)
				7.3
				}

{
V(S)=N(R)
	{
	V(S)=N(R)
	STOP T200
	START T203
	V(A)=N(R)
	7.0
	}
		{
		V(S)=N(R)
		STOP T200
		START T203
		V(A)=N(R)
		7.1
		}
			{
			V(S)=N(R)	

STOP T200 START T203 V(A)=N(R) 7.2 } V(S)=N(R) STOP T200 START T203 V(A)=N(R) 7.3 }	{				

{
RR command
P=1
N(R) error
| |
}
TX RR F=1
MDL-ERR-IND(J)
RC =0
TX SABME P=1
RESTART T200
5.1
}
TX RNR F=1
MDL-ERR-IND(J)
RC =0
TX SABME P=1
RESTART T200
5.1
}
TX RR F=1
MDL-ERR-IND(J)
RC =0
TX SABME P=1
RESTART T200
5.1
}
TX RNR F=1
MDL-ERR-IND(J)
RC =0
TX SABME P=1
RESTART T200
5.1
}
	{	{
		{
		{			

{
RR command
P=0
N(R) error
| |
}
MDL-ERR-IND(J)
RC =0
TX SABME P=1
RESTART T200
5.1
}
	{				

RR response F=0 N(R) error					
RR response F=1 N(R) error					

Table D-3/Q.921 (4 of 10) [5T21.921], p.

H.T. [6T21.921]

TABLE D-3/Q.921 (5 of 10)
{
State transition table: receiving REJ supervisory frame with correct format, clearance of timer recovery if there is F = 1 only
}

BASIC STATE	TIMER RECOVERY

TRANSMITTER CONDITION	NORMAL	NORMAL	NORMAL	NORMAL	PEER
RECEIVER CONDITION	NORMAL	REJ RECOVERY	OWN REC BUSY	REJ and own REC busy	NO
STATE NUMBER	8.0	8.1	8.2	8.3	

{
REJ command
P=1
V(A) N(R) V(S)
}
	TX RR F=1 V(A)=N(R)		TX RNR F=1 V(A)=N(R)		TX RR F=1

{
REJ command
P=0
V(A) N(R) V(S)
}
	V(A)=N(R)				V(A)=N(R)

{
REJ response
F=0
V(A) N(R) V(S)
}					

{
REJ response
F=1
V(A) N(R) V(S)
| |
}
	{
	V(S)=V(A)=N(R)
	STOP T200
	START T203
	7.0				

}	{ V(S)=V(A)=N(R) STOP T200 START T203 7.1				
}	{ V(S)=V(A)=N(R) STOP T200 START T203 7.2				
}	{ V(S)=V(A)=N(R) STOP T200 START T203 7.3				
}	{ V(S)=V(A)=N(R) STOP T200 START T203 7.0				
}	{ V(S)=V(A)=N(R) STOP T200 START T203 7.1				
}	{ V(S)=V(A)=N(R) STOP T200 START T203 7.2				
}	{ V(S)=V(A)=N(R) STOP T200 START T203				

7.3 }					

{
REJ command
P=1
N(R) error
| |
}
	{
	TX RR F=1
	MDL-ERR-IND(J)
	RC =0
	TX SABME P=1
	RESTART T200
	5.1
	}
		{
		TX RNR F=1
		MDL-ERR-IND(J)
		RC =0
		TX SABME P=1
		RESTART T200
		5.1
		}
			{
			TX RR F=1
			MDL-ERR-IND(J)
			RC =0
			TX SABME P=1
			RESTART T200
			| |
			5.1
			}
				{
				TX RNR F=1
				MDL-ERR-IND(J)
				RC =0
				TX SABME P=1
				RESTART T200
				| |
				5.1
				}	

{
REJ command
P=0
N(R) error
| |
}
	{
	MDL-ERR-IND(J)
	RC =0
	TX SABME P=1
	RESTART T200
	5.1
	}				

REJ response F=0 N(R) error					
REJ response F=1 N(R) error					

Table D-3/Q.921 (5 of 10) [6T21.921], p.

H.T. [7T21.921]

TABLE D-3/Q.921 (6 of 10)
{
State transition table: receiving RNR supervisory frame with correct format, clearance of timer recovery if there is F = 1 only
}

BASIC STATE	TIMER RECOVERY

TRANSMITTER CONDITION RECEIVER CONDITION STATE NUMBER	NORMAL NORMAL 8.0	NORMAL REJ RECOVERY 8.1	NORMAL OWN REC BUSY 8.2	NORMAL REJ and own REC 8.3

{
RNR command
P=1
V(A) N(R) V(S)
}
	TX RR F=1 V(A)=N(R) 8.4	TX RR F=1 V(A)=N(R) 8.5	TX RNR F=1 V(A)=N(R) 8.6	TX RNR F=1 V(A)=N

{
RNR command
P=0
V(A) N(R) V(S)
}
	V(A)=N(R) 8.4	V(A)=N(R) 8.5	V(A)=N(R) 8.6	V(A)=N(R) 8.7

{
RNR response
F=0
V(A) N(R) V(S)
}				

{
RNR response
F=1
V(A) N(R) V(S)
| |
}
	{
	V(S)=N(R)
	RESTART T200
	V(A)=N(R)
	7.4			

}	{ V(S)=N(R) RESTART T200 V(A)=N(R) 7.5			
}	{ V(S)=N(R) RESTART T200 V(A)=N(R) 7.6			
}	{ V(S)=N(R) RESTART T200 V(A)=N(R) 7.7			
}	{ V(S)=N(R) RESTART T200 V(A)=N(R) 7.4			
}	{ V(S)=N(R) RESTART T200 V(A)=N(R) 7.5			
}	{ V(S)=N(R) RESTART T200 V(A)=N(R) 7.6			
}	{ V(S)=N(R) RESTART T200 V(A)=N(R)			

7.7 }				

{
RNR command
P=1
N(R) error
| |
}
	{
	TX RR F=1
	MDL-ERR-IND(J)
	RC =0
	TX SABME P=1
	RESTART T200
	5.1
	}
		{
		TX RNR F=1
		MDL-ERR-IND(J)
		RC =0
		TX SABME P=1
		RESTART T200
		5.1
		}
			{
			TX RR F=1
			MDL-ERR-IND(J)
			RC =0
			TX SABME P=1
			RESTART T200
			5.1
			}
				{
				TX RNR F=1
				MDL-ERR-IND(J)
				RC =0
				TX SABME P=1
				RESTART T200
				5.1
				}

{
RNR command
P=0
N(R) error
| |
}
	{
	MDL-ERR-IND(J)
	RC =0
	TX SABME P=1
	RESTART T200
	5.1
	}			

RNR response F=0 N(R) error				
RNR response F=1 N(R) error				

Table D-3/Q.921 (6 of 10) [7T21.921], p.

H.T. [8T21.921]

TABLE D-3/Q.921 (7 of 10)

{

State transition table: receiving I command frame with correct format acknowledging all outstanding I frames or containing an N(R) which satisfies V(A) | | (R) | | (S); no clearance of timer recovery

}

BASIC STATE	TIMER RECOVERY

TRANSMITTER CONDITION	NORMAL	NORMAL	NORMAL	NORMAL	PEER REC BUSY	PEER RE
RECEIVER CONDITION	NORMAL	REJ RECOVERY	OWN REC BUSY	REJ and own REC busy	NORMAL	REJ REC
STATE NUMBER	8.0	8.1	8.2	8.3	8.4	8.
{						
I command P=1						
N(S) = V(R)						
N(R) = V(S)						
| |						
}	{					
V(R)=V(R)+1						
DL=DATA=IND						
TX RR F=1						
V(A)=N(R)						
| |						
}	{					
V(R)=V(R)+1						
DL=DATA=IND						
TX RR F=1						
V(A)=N(R)						
8.0						
}	{					
‘‘DISCARD’’						
TX RNR F=1						
V(A)=N(R)						
| |						
| |						
}		{				
V(R)=V(R)+1						
DL-DATA-IND						
TX RR F=1						
V(A)=N(R)						
| |						
}	{					
V(R)=V(R)+1						
DL-DATA-IND						
TX RR F=1						
V(A)=N(R)						
8.4						
}	{					
‘‘DISCARD’’						
TX RNR F=1						
V(A)=N(R)						
| |						
| |						
}						
{						
I command P=0						
N(S) = V(R)						
N(R) = V(S)						
| |						
}	{					
V(R)=V(R)+1						
DL=DATA=IND						
TX ACK						
V(A)=N(R)						
| |						
}	{					
V(R)=V(R)+1						
DL=DATA=IND						
TX ACK						
V(A)=N(R)						
8.0						

}
{
‘‘DISCARD’’
| |
V(A)=N(R)
| |
| |
}
{
V(R)=V(R)+1
DL-DATA-IND
TX RR F=0
V(A)=N(R)
| |
}
{
V(R)=V(R)+1
DL-DATA-IND
TX RR F=0
V(A)=N(R)
8.4
}
{
‘‘DISCARD’’
V(A)=N(R)
| |
| |
| |
}

{
I command P=1
N(S) ≠ V(R)
N(R) = V(S)
| |
}
{
‘‘DISCARD’’
TX REJ F=1
V(A)=N(R)
8.1
}
{
‘‘DISCARD’’
TX RR F=1
V(A)=N(R)
| |
}
{
‘‘DISCARD’’
TX RNR F=1
V(A)=N(R)
| |
}
{
‘‘DISCARD’’
TX REJ F=1
V(A)=N(R)
8.5
}
{
‘‘DISCARD’’
TX RR F=1
V(A)=N(R)
| |
}
{
‘‘DISCARD’’
TX RNR F=1
V(A)=N(R)
| |
}

{

I command P=0
N(S) ≠ V(R)
N(R) = V(S)
| |
} {
‘‘DISCARD’’
TX REJ F=0
V(A)=N(R)
8.1
} {
‘‘DISCARD’’
V(A)=N(R)
| |
| |
} {
‘‘DISCARD’’
TX REJ F=0
V(A)=N(R)
8.5
} {
‘‘DISCARD’’
V(A)=N(R)
| |
| |
}

{
I command P=1
N(S) = V(R)
V(A) | | (R) | | (S)
| |
} {
V(R)=V(R)+1
DL=DATA=IND
TX RR F=1
V(A)=N(R)
| |
} {
V(R)=V(R)+1
DL=DATA=IND
TX RR F=1
V(A)=N(R)
8.0
} {
‘‘DISCARD’’
TX RNR F=1
V(A)=N(R)
| |
| |
} {
V(R)=V(R)+1
DL-DATA-IND
TX RR F=1
V(A)=N(R)
| |
} {
V(R)=V(R)+1
DL-DATA-IND
TX RR F=1
V(A)=N(R)
8.4
} {
‘‘DISCARD’’

TX RNR F=1
V(A)=N(R)
| |
| |
}

lw(54p) | lw(36p) | lw(36p) | lw(36p) | cw(36p) | lw(36p) | lw(36p) | lw(36p) | cw(36p) .
{
I command P=0
N(S) = V(R)
V(A) | | (R) | | (S)
| |
} {
V(R)=V(R)+1
DL=DATA=IND
TX ACK
V(A)=N(R)
| |
} {
V(R)=V(R)+1
DL=DATA=IND
TX ACK
V(A)=N(R)
8.0
} {
‘‘DISCARD’’
V(A)=N(R)
| |
| |
| |
} {
V(R)=V(R)+1
DL-DATA-IND
TX RR F=0
V(A)=N(R)
| |
} {
V(R)=V(R)+1
DL-DATA-IND
TX RR F=0
V(A)=N(R)
8.4
} {
‘‘DISCARD’’
V(A)=N(R)
| |
| |
| |
}

lw(54p) | lw(36p) | lw(36p) | lw(36p) | cw(36p) | lw(36p) | lw(36p) | lw(36p) | cw(36p) .
{
I command P=1
N(S) ≠ V(R)
V(A) | | (R) | | (S)
| |
} {
‘‘DISCARD’’
TX REJ F=1
V(A)=N(R)
8.1
} {
‘‘DISCARD’’
TX RR F=1

V(A)=N(R) | | }	{					
‘‘DISCARD’’ TX RNR F=1 V(A)=N(R) | | }		{				
‘‘DISCARD’’ TX REJ F=1 V(A)=N(R) 8.5 }	{					
‘‘DISCARD’’ TX RR F=1 V(A)=N(R) | | }	{					
‘‘DISCARD’’ TX RNR F=1 V(A)=N(R) | | }						

lw(54p) | lw(36p) | lw(36p) | lw(36p) | cw(36p) | lw(36p) | lw(36p) | lw(36p) | cw(36p) .
{
I command P=0
N(S) ≠ V(R)
V(A) | | (R) | | (S)
| |
}
	{					

‘‘DISCARD’’ TX REJ F=0 V(A)=N(R) 8.1 }	{					
‘‘DISCARD’’ V(A)=N(R) | | | | }			{			
‘‘DISCARD’’ TX REJ F=0 V(A)=N(R) 8.5 }	{					
‘‘DISCARD’’ V(A)=N(R) | | | | }						

Table D-3/Q.921 (7 of 10) [8T21.921], p.

TABLE D-3/Q.921 (8 of 10)
{
State transition table: receiving I command frame with correct format containing an N(R) which satisfies V(A) | | (R) | | (S), or an N(R) error

}

BASIC STATE	TIMER RECOVERY

TRANSMITTER CONDITION	NORMAL	NORMAL	NORMAL	NORMA
RECEIVER CONDITION	NORMAL	REJ RECOVERY	OWN REC BUSY	REJ and own R
STATE NUMBER	8.0	8.1	8.2	8.3
{				
I command P=1				
N(S) = V(R)				
V(A)=N(R) | | (S)				
}	{			
V(R)=V(R)+1				
DL=DATA=IND				
TX RR F=1				
| |				
}	{			
V(R)=V(R)+1				
DL=DATA=IND				
TX RR F=1				
8.0				
}	{			
‘‘DISCARD’’				
TX RNR F=1				
| |				
| |				
}		{		
V(R)=V(R)+1				
DL-DATA-IND				
TX RR F=1				
| |				
}	{			
V(R)=V(R)+1				
DL-DATA-IND				
TX RR F=1				
8.4				
}	{			
‘‘DISCARD’’				
TX RNR F=1				
| |				
| |				
}				
{				
I command P=0				
N(S) = V(R)				
V(A)=N(R) | | (S)				
}	{			
V(R)=V(R)+1				
DL=DATA=IND				
TX ACK				
| |				
}	{			
V(R)=V(R)+1				
DL=DATA=IND				
TX ACK				
8.0				
}	{			
‘‘DISCARD’’				
| |				
| |				
| |				
}		{		
V(R)=V(R)+1				
DL-DATA-IND				
TX RR F=0				
| |				

}	{		
V(R)=V(R)+1 DL-DATA-IND TX RR F=0 8.4			
}	{		
‘‘DISCARD’’ | | | | | |			
}			
{			
I command P=1 N(S) ≠ V(R) V(A)=N(R) | | (S)			
}	‘‘DISCARD’’ TX REJ F=1 8.1	‘‘DISCARD’’ TX RR F=1 | |	{
‘‘DISCARD’’ TX RNR F=1 | |			
}		‘‘DISCARD’’ TX REJ F=1 8.5	‘‘DISCARD’’ TX RR F=1 | | {
‘‘DISCARD’’ TX RNR F=1 | |			
}			
{			
I command P=0 N(S) ≠ V(R) V(A)=N(R) | | (S)			
}	‘‘DISCARD’’ TX REJ F=0 8.1	‘‘DISCARD’’ | | | |	
{			
I command P=1 N(S) = V(R) N(R) error | |			
}	{		
V(R)=V(R)+1 DL=DATA=IND TX RR F=1 MDL-ERR-INJ(J) RC = 0 TX SABME P=1 RESTART T200 5.1			
}		{	
‘‘DISCARD’’ TX RNR F=1 MDL-ERR-INJ(J) RC = 0 TX SABME P=1 RESTART T200 | |			
5.1			
}		{	
V(R)=V(R)+1 DL=DATA=IND TX RR F=1 MDL-ERR-INJ(J) RC = 0 TX SABME P=1 RESTART T200 5.1			
}		{	

‘‘DISCARD’’
TX RNR F=1
MDL-ERR-INJ(J)
RC = 0
TX SABME P=1
RESTART T200
| |
5.1
}

{
I command P=0
N(S) = V(R)
N(R) error
| |
}

{
V(R)=V(R)+1
DL=DATA=IND
MDL-ERR-INJ(J)
RC = 0
TX SABME P=1
RESTART T200
5.1
}

{
‘‘DISCARD’’
MDL-ERR-INJ(J)
RC = 0
TX SABME P=1
RESTART T200
| |
5.1
}

{
V(R)=V(R)+1
DL=DATA=IND
MDL-ERR-INJ(J)
RC = 0
TX SABME P=1
RESTART T200
5.1
}

{
‘‘DISCARD’’
MDL-ERR-INJ(J)
RC = 0
TX SABME P=1
RESTART T200
| |
5.1
}

{
I command P=1
N(S) ≠ V(R)
N(R) error
| |
}

{
‘‘DISCARD’’
TX REJ F=1
MDL-ERR-IND(J)
RC = 0
TX SABME P=1
RESTART T200
5.1
}

{
‘‘DISCARD’’

TX RR F=1
MDL-ERR-IND(J)
RC = 0
TX SABME P=1
RESTART T200
5.1
} {
‘‘DISCARD’’
TX RNR F=1
MDL-ERR-IND(J)
RC = 0
TX SABME P=1
RESTART T200
5.1
} {
‘‘DISCARD’’
TX REJ F=1
MDL-ERR-IND(J)
RC = 0
TX SABME P=1
RESTART T200
5.1
} {
‘‘DISCARD’’
TX RR F=1
MDL-ERR-IND(J)
RC = 0
TX SABME P=1
RESTART T200
5.1
} {
‘‘DISCARD’’
TX RNR F=1
MDL-ERR-IND(J)
RC = 0
TX SABME P=1
RESTART T200
5.1
}

---

lw(54p) | lw(36p) | lw(36p) | lw(36p) | cw(36p) | lw(36p) | lw(36p) | lw(36p) | cw(36p) .
{
‘‘DISCARD’’
TX REJ F=0
MDL-ERR-IND(J)
RC = 0
TX SABME P=1
RESTART T200
5.1
} {
‘‘DISCARD’’
TX REJ F=0
MDL-ERR-IND(J)
RC = 0
TX SABME P=1
RESTART T200
5.1
} {
‘‘DISCARD’’
MDL-ERR-IND(J)
RC = 0
TX SABME P=1
RESTART T200

| | 5.1 }				{
‘‘DISCARD’’ TX REJ F=0 MDL-ERR-IND(J) RC = 0 TX SABME P=1 RESTART T200 5.1 }	{			
‘‘DISCARD’’ MDL-ERR-IND(J) RC = 0 TX SABME P=1 RESTART T200 | | 5.1 }				

Table D-3/Q.921 (8 of 10) [9T21.921], p.

TABLE D-3/Q.921 (9 of 10)
{
State transition table: internal events (expiry of timers, receiver busy condition); initiation of a re-establishment procedure if the value of the retransmission count variable is equal to N200

}

BASIC STATE	TIMER RECOVERY

TRANSMITTER CONDITION	NORMAL	NORMAL	NORMAL	NORMAL	PEER REC BUSY
RECEIVER CONDITION	NORMAL	REJ RECOVERY	OWN REC BUSY	REJ and own REC busy	NORMAL
STATE NUMBER	8.0	8.1	8.2	8.3	8.4
{ T200 TIMEOUT RC < | 200 V(A) < V(S) | |					
}	{				
either V(S)=V(S)—1 TX I P=1 V(S)=V(S)+1 or TX RR P=1 then RC=RC+1 START T200					
}		{			
either V(S)=V(S)—1 TX I P=1 V(S)=V(S)+1 or TX RNR P=1 then RC=RC+1 START T200					
}		{			
TX RR P=1 RC=RC+1 START T200 | | | | | | | | | |					
}	{				
TX RNR P=1 RC=RC+1 START T200 | | | | | | | | | |					
}					
{ T200 TIMEOUT RC < | 200 V(A) = V(S)					
}	{				
TX RR P=1 RC = RC+1 START T200					
}		{			
TX RR P=1 RC = RC+1 START T200					
}					
{ T200 TIMEOUT					

RC = N200 | | } MDL-ERR-IND(I) RC=0 TX SABME P=1 START T200 5.1 }	{				
T203 TIME-OUT	/	/	/	/	/
SET OWN RECEIVER BUSY (Note)	TX RNR F=0 8.2	TX RNR F=0 8.3	—	—	TX RNR F=0 8.6

{
CLEAR
OWN RECEIVER BUSY (Note)
}
	—	—	TX RR F=0 8.0	TX RR F=0 8.1	—

Note — These signals are generated outside the procedures specified in this state transition table, and may be generated by the connection management entity.

Table D-3/Q.921 (9 of 10) [10T21.921], p.

H.T. [11T21.921]

{
TABLE D-3/Q.921 (10 of 10)
}
{
State transition table: receiving frame with incorrect format or frame not implemented
}

BASIC STATE	TIMER RECOVERY

TRANSMITTER CONDITION RECEIVER CONDITION STATE NUMBER	NORMAL NORMAL 8.0	NORMAL REJ RECOVERY 8.1	NORMAL OWN REC BUSY 8.2	NORMAL REJ and own REC busy 8.3	PEER REC BUSY NORMAL 8.4
SABME incorrect length | | MDL-ERR-IND(N) RC = 0 TX SABME P=1 RESTART T200 5.1	{				
DISC incorrect length					
UA incorrect length					
DM incorrect length					
FRMR incorrect length					

{
Supervisory frame
RR, REJ, RNR
incorrect length
}					

N201 error | | MDL-ERR-IND(0) RC = 0 TX SABME P=1 RESTART T200 5.1	{				

{
Undefined command and response frames
| |
}
MDL-ERR-IND(L)
RC = 0
TX SABME P=1
RESTART T200
5.1
	{				

I field not permitted | | MDL-ERR-IND(M) RC = 0 TX SABME P=1 RESTART T200 5.1	{				

Table D-3/Q.921 (10 of 10) [11T21.921], p.

APPENDIX I
(to Recommendation Q.921)

Retransmission of REJ response frames

I.1 *Introduction*

This appendix describes an optional procedure which may be used to provide a reject retransmission procedure.

I.2 *Procedure*

This optional reject retransmission procedure can supplement the Q.921 LAPD protocol by defining a new variable for multiple frame operation (§ 3.5.2), and by modifying the N(S) sequence error exception condition reporting and recovery (§ 5.8.1).

I.2.1 *Recovery state variable V(M)*

Each point-to-point data link entity may have an associated V(M) when using I frame commands and supervisory frame commands/responses. V(M) denotes the sequence number of the last frame received which caused an N(S) sequence error condition. V(M) can take on the value 0 to 127 and may be used to determine if another REJ response frame should be sent on receipt of an N(S) sequence error while in the REJ exception condition.

I.2.2 *N(S) sequence error supplementary procedure*

The first three paragraphs of § 5.8.1, N(S) sequence error, apply. The remainder of the section if as follows:

The REJ frame is used by a receiving data link layer entity to initiate an exception recovery (retransmission) following the detection of an N(S) sequence error. The receiving data link entity shall set V(M) to the N(S) sequence number which caused the N(S) sequence error condition.

Only one REJ exception condition for a given direction of information transfer shall be established at a time [that is, all REJ frames must have the same N(R) value until the REJ reception is cleared].

A data link layer entity receiving an REJ command or response shall initiate sequential transmission (retransmission) of I frames starting with the I frame indicated by the N(R) contained in the REJ frame.

A REJ exception is cleared when the requested I frame is received or when SABME, or DISC is received.

If an N(S) sequence error exception occurs when the receiving data link layer entity is in the REJ exception condition, then check the N(S) of the received frame to see if the data link layer entity which received the REJ frame has retransmitted in response to the REJ frame [i.e., is N(S) within the range V(R) + 1 N(S) V(M)]. If the N(S) of the received frame is within the above range, then send another REJ response frame, issue an MDL-ERROR-INDICATION primitive to the connection management entity, and set V(M) equal to N(S). The transmitting side will not need to wait for timer T200 to expire before it can retransmit the lost frame.

If an N(S) sequence error occurs when the receiving data link layer entity is in the REJ exception condition, and it cannot be determined if the data link layer entity which received the REJ frame has retransmitted in response to that frame [i.e., if N(S) > V(M)], then set V(M) equal to the N(S) of the received frame.

Blanc

APPENDIX II
(to Recommendation Q.921)

Occurrence of
MDL-ERROR-INDICATION

within the basic states and actions

to be taken by the management entity

II.1 *Introduction*

Table II-1/Q.921 gives the error situations in which the MDL-ERROR-INDICATION primitive will be generated. This primitive notifies the data link layer's connection management entity of the occurred error situation. The associated error parameter contains the error code that describes the unique error conditions. Table II-1/Q.921 also identifies the associated connection management actions to be taken from the network and the user side, based on the types of error conditions reported.

This appendix does not incorporate the retransmission of REJ response frames described in Appendix I.

II.2 *Layout of Table II-1/Q.921*

The ''Error code'' column gives the identification value of each error situation to be included as a parameter with the MDL-ERROR-INDICATION primitive.

The column entitled ''Error condition'' together with the ''Affected states'' describes unique protocol error events and the basic state of the data link layer entity at the point that the MDL-ERROR-INDICATION primitive is generated.

For a given error condition, the column entitled ''Network management action'' describes the preferred action to be taken by the network management entity.

The column entitled ''User management action'' describes the preferred action to be taken by the user side management entity on a given error condition.

II.3 *Preferred management actions*

The various preferred layer management actions on an error situation may be described as one of the following:

a) Error log

This suggests that the network side connection management entity has the preferred action of logging the event into an error counter. The length and the operation of the counter mechanisms for the error situations is implementation dependent.

b) TEI check

This means that the network side layer management entity invokes the TEI check procedure.

c) TEI verify

This means that the user side layer management entity may optionally invoke a TEI verify request procedure that asks the network side layer management entity to issue a TEI check procedure.

d) TEI remove

This means that the user side layer management entity may directly remove its TEI value from service.

In most of the described error situations, there is either no action to be taken on the user side layer management or the action to be taken is implementation dependent, as Table II-1/Q.921 shows. ''Implementation dependent'' means that it is optional whether the

user side layer management has incorporated any form of error counter to log (store) the reported event. If action is taken, the layer management has to take into account that the data link layer will have initiated a recovery procedure.

Blanc

H.T. [T22.921]

TABLE II-1/Q.921

Management Entity Actions for MDL Error Indications

Error Type	*Error Code*	*Error Condition*	{		
Affected states (See Note 1) } *Network Management Action* } *User Management Action* }	{ {				
{					
Unable to convert table					

Table II-1/Q.921 [T22.921] + Notes, p.

Blanc

APPENDIX III
(to Recommendation Q.921)

Optional basic access deactivation procedures

III.1 *Introduction*

This appendix provides one example of a deactivation procedure which can be used by the network side system management to control deactivation of the access. Figure III-1/Q.921 provides a conceptual model of the interactions which are required for this deactivation procedure.

Figure III-1/Q.921, p.

III.2 *Description of the Conceptual Model*

The monitor function uses layer 2 activity as the basis for establishing whether deactivation of the access can take place. The signal INFORMATION is used to report the layer 2 activity in the following manner:

— INFORMATION (FREE) indicates that there is no data link connection in the multiple-frame mode of operation;

— INFORMATION (IN USE) indicates that there is at least one data link connection in the mode-setting or multiple-frame mode of operation; and

— INFORMATION (UNIT DATA) indicates that a UI frame is about to be transmitted, or has just been received.

Within the data link layer entity the DL-ESTABLISH-REQUEST/INDICATION primitives and DL-RELEASE-INDICATION/CONFIRM mark the duration of the multiple-frame mode of operation, and the MDL/DL/UNIT DATA-REQUEST/INDICATION primitives mark the transmission and reception of UI frames.

A signal Status is used to represent the ability of higher layers to enable or disable the deactivation procedures:

— STATUS (ENABLE) deactivation procedures enabled; and

— STATUS (DISABLE) deactivation procedures disabled.

The MPH-DEACTIVATE-REQ, MPH-DEACTIVATE-IND and MPH-ACTIVATE-IND primitives are used as described in § 4. The definition and usage of these primitives are also described in Recommendation I.430 [4] which specifies layer 1.

Since, in Recommendation I.430, the usage of the MPH-DEACTIVATE-IND primitive is an implementation option, two cases of deactivation are described below.

§ III.3 provides a description of the deactivation procedure when the MPH-DEACTIVATE-IND primitive is delivered to the system management entity.

§ III.4 provides a description of the deactivation procedure when the MPH-DEACTIVATE-IND primitive is not delivered to the system management entity.

Note — These procedures require that all layer 3 entities making use of the acknowledged information transfer service, must release the data link connection at an appropriate point after the completion of the information transfer.

III.3 *Deactivation procedure with MPH-DEACTIVATE-IND*

This deactivation procedure makes use of the MPH-DEACTIVATE-IND primitive to provide an option of layer 1 implementation.

Figure III-2/Q.921 provides a state transition diagram of the deactivation procedure with the MPH-DEACTIVATE-IND primitive.

This deactivation procedure can be represented by six states:

State 1 Information transfer not available and free;

(*No info xfer and free*)

State 2 Information transfer available and free;

(*Info xfer and free*)

State 3 Information transfer available and in use;

(*Info xfer and in use*)

State 4 Information transfer not available and in use;

(*No info transfer and in use*)

State 5 Information transfer interrupted and free;

(*Info interrupted and free*)

State 6 Information transfer interrupted and in use;

(*Info interrupted and in use*)

These six states are described as follows:

— State 1 represents the state where the access is assumed to be deactivated and no data link connections are in a mode setting or multiple-frame mode of operation.

— State 2 represents the state where the access is activated and no data link connection is in a mode setting or multiple-frame mode of operation. Timer TM01 is running, and upon its expiry, if deactivation is enabled, then an MPH-DEACTIVATE-REQ primitive may be issued to layer 1. The access is then assumed to be deactivated.

— State 3 represents the state where the access is activated and at least one data link connection is in a mode setting or multiple-frame mode of operation.

— State 4 represents the state where the access is regarded as being in an transient state (neither deactivated nor activated) and at least one data link connection is in a mode setting or multiple-frame mode of operation. [This state can be entered, for example, due to the arrival of an INFORMATION (IN USE) signal before an MPH-ACTIVATE-IND primitive.]

— State 5 represents the state where the access is regarded as being in a transient state (neither deactivated nor activated) and no data link connection is in a mode setting or multiple-frame mode of operation. Timer TM01 is running and upon its expiry, if deactivation is enabled, then an MPH-DEACTIVATE-REQ primitive will be issued to layer 1. The access is assumed to be deactivated.

— State 6 represents the state where the access is regarded as being in the transient state (neither deactivated nor activated) and at least one data link connection is in a mode setting or multiple frame mode of operation.

Timer TM01 is started whenever state 2 is entered:

— on receipt of an MPH-ACTIVATE-IND primitive in state 1; and

— on receipt of an INFORMATION (FREE) signal in state 3.

Timer TM01 is started whenever state 5 is entered:

— on receipt of an INFORMATION (FREE) signal in state 6.

Timer TM01 is restarted in states 2 and 3 when:

— TM01 expires while deactivation is disabled by the receipt of a STATUS (DISABLE) signal; and

— an INFORMATION (UNIT DATA) signal is received in order to allow sufficient time for current and further unacknowledged information transfer.

Timer TM01 has a value of ten seconds at the network side.

Blanc

Figure III-2/Q.921, p.

III.4 *Deactivation procedure without MPH-DEACTIVATE-IND*

This deactivation procedure does not make use of the MPH-DEACTIVATE-IND primitive to provide an option of layer 1 implementation. Thus this procedure can be represented by only four states, i.e. state 1, state 2, state 3, and state 4. States 5 and 6 have disappeared.

Figure III-3/Q.921 provides a state transition diagram of this deactivation procedure without the MPH-DEACTIVATE-IND primitive.

Figure III-3/Q.921, p.

APPENDIX IV
(to Recommendation Q.921)

Automatic negotiation of data link layer parameters

IV.1 *General*

Each data link layer entity has an associated data link connection management entity. The data link connection management entity has the responsibility for initializing the link parameters necessary for correct peer-to-peer information transport.

The method of initialization of the parameters follows one of the two methods below:

— initialization to the default values as specified in § 5.9; or

— initialization based on the values supplied by its peer entity.

The latter method utilizes the parameter negotiation procedure described in this appendix. Typically, after the assignment of a TEI value to the management entity, the data link connection management entity is notified by its layer management entity that parameter initialization is required.

The data link connection management entity will invoke the peer-to-peer notification procedure the data link connection management entity will notify the layer management entity that parameter initialization has occurred, and the layer management entity will issue the MDL-ASSIGN-REQUEST.

IV.2 *Parameter initialization*

The parameter initialization procedure may invoke either the internal initialization procedure or the automatic notification of data link parameter procedure.

IV.3 *Internal parameter initialization*

When the layer management entity notifies the connection management entity of TEI assignment, the connection management entity shall initialize the link parameters to the default values and notify the layer management of task completion.

IV.4 *Automatic notification of data link layer parameter values*

For each data link layer an exchange of certain data link layer parameters may take place between the peer data link connection management entities before entering the *TEI-assigned* state. This exchange may be initiated after acquiring a TEI, that is, after:

— receipt of a DL-ESTABLISH-REQUEST or a DL-UNIT DATA-REQUEST primitive following a power-up condition associated with non-automatic TEI user equipment.

— receipt of the Identity assigned response for automatic TEI assignment user equipment. This message contains the TEI received by the layer management entity.

The data link connection management entity, following assignment of a TEI from the layer management entity, shall issue an XID command with the P bit set to 0 and containing the parameter message shown in Figure IV-1/Q.921, and start the connection management timer TM20.

The I field of the XID command frame shall reflect the parameters desired for future communications across this data link layer connection.

The peer data link connection management entity, upon receipt of this XID command frame, shall transmit an XID response with the F bit set to 0 containing the list of parameter values that the peer can support.

If the data link connection management entity receives the above XID response prior to expiry of timer TM20, it shall stop the timer, and shall notify the layer management entity of a successful parameter exchange. However, if timer TM20 expires before receiving the XID response, the data link connection management entity shall retransmit the XID command, increment the retransmission counter and restart timer TM20. This retransmission process is repeated if timer TM20 expires again. Should the retransmission counter equal NM20, or an XID response frame with a zero length I field be received, the data link connection management entity shall issue an indication to the layer management entity and initialize the parameters to the default values. The layer management entity may log this condition and then issue the MDL-ASSIGN-REQUEST primitive to the data link layer.

The timer TM20 is set to 2.5 seconds and NM20 is set to 3.

Figure IV-3/Q.921 [T23.921], p.

ABBREVIATIONS AND ACRONYMS USED IN RECOMMENDATION Q.921

Abbreviation of acronym	*Meaning*
Ai	Action indicator
ASP	Assignment source point
CEI	Connection endpoint identifier
CES	Connection endpoint suffix
C/R	Command/response field bit
DISC	Disconnect
DL-	Communication between Layer 3 and data link layer
DLCI	Data link connection identifier
DM	Disconnected mode
EA	Extended address field bit
ET	Exchange termination
FCS	Frame check sequence
FRMR	Frame reject
I	Information
ID	Identity
ISDN	Integrated Services Digital Network
L1	Layer 1
L2	Layer 2
L3	Layer 3
LAPB	Link access procedure — Balanced
LAPD	Link access procedure on the D-channel
M	Modifier function bit
MDL-	Communication between management entity and data link layer
MPH-	Communication between system management and physical layer
N(R)	Receive sequence number
N(S)	Send sequence number
P/F	Poll/Final bit

PH- Communication between data link layer and physical layer

RC Retransmission counter

REC Receiver

REJ Reject

Ri Reference number

RNR Receive not ready

RR Receive ready

S Supervisory

S Supervisory function bit

SABME Set asynchronous balanced mode extended

SAP Service access point

SAPI Service access point identifier

TE Terminal equipment

TEI Terminal endpoint identifier

TX Transmit

U Unnumbered

UA Unnumbered acknowledgement

UI Unnumbered information

V(A) Acknowledge state variable

V(M) Recovery state variable

V(R) Receive state variable

V(S) Send state variable

XID Exchange identification

References

[1] CCITT Recommendation Q.920 (I.440), *ISDN user-network interface data link layer — General aspects* .

[2] CCITT Recommendation Q.930 (I.450), *ISDN user-network interface layer 3 — General aspects* .

[3] CCITT Recommendation Q.931 (I.451), *ISDN user-network interface layer 3 specification* .

[4] CCITT Recommendation I.430, *Basic user-network interface layer 1 specification* .

[5] CCITT Recommendation I.431, *Primary rate user-network interface layer 1 specification* .

[6] CCITT Recommendation X.25, *Interface between data terminal equipment (DTE) and data circuit terminating equipment (DCE) for terminals operating in the packet mode and connected to public data networks by dedicated circuit* .

A different acronym has to be found for Supervisory function bit.

MONTAGE: PAGE PAIRE = PAGE BLANCHE